AN EVALUATION OF

TEXT PROCESSING SYSTEMS

by

DAVID DANIEL URBANI

B.S. Drexel University
(1969)

M.S. Carnegie Mellon University
(1971)

SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF

SCIENCE

at the

MASSACHUSETTS INSTITUTE OF

TECHNOLOGY

June, 1973

Signature of Author..
Alfred P. Sloan School of Management, May 11, 1973

Certified by..
Thesis Supervisor

Accepted by..
Chairman, Departmental Committee on Graduate Students

An Evaluation of Text Processing Systems

David Urbani

Submitted to the Alfred P.Sloan School of Management on May 11, 1973 in partial fulfillment of the requirements for the degree of Master of Science.

ABSTRACT

Until recently, computers have been used almost exclusively for data processing functions. The payoffs for automating accounting procedures and scientific calculational procedures was large enough to justify the relatively expensive early computer systems. However, as economies of scale and new technology lower the effective cost of computer systems, new and more imaginative applications are being developed. Text processing is one of these applications, and it is to this subject that this thesis is directed.

Text processing is the application of computer technology to the production of textual information. There is a wide range of systems and capabilities available today which can be divided into three groups: automatic typewriters, mini-computer supported terminals, and multi-purpose computer supported terminals. Several systems of each type were evaluated and compared. Also, several common, and some not so common, uses of text processing were explored. Since each system has its own advantages and disadvantages, the matching of application requirements to system capabilities is the best method of system selection.

To determine the productivity increase made possible by text processing systems, three sources were evaluated: the production of this thesis, various industry reports, and a test conducted on ten subjects. The productivity increase realized all cases were significant, and ranged from 50 to 1000 per cent.

Overall, text processing has been shown to be a viable and cost effective method of producing documents in the business environment.

Thesis Supervisor: Stuart A. Madnick
Title: Assistant Professor of Management

ACKNOWLEDGEMENTS

I would like to express my gratitude, first of all, to my thesis supervisor, Professor Stuart A. Madnick, who gave me much of his time and many sources of information, especially in the early stages of this thesis.

I would also like to thank all of the individuals who provided me with much of the information contained in this thesis, particularly: Dan Diamond at Index Systems, Red Chellis at Camp, Dresser & Mckee, Jim Jackson at Engineering Computers International, Ed Mosher at IBM, Cambridge Scientific Center, Don Nordbeck at Stone & Webster, Nat Rochester at IBM, Cambridge Scientific Center, and Andries van Dam and associates at Brown University.

I would also like to thank all of those people who volunteered 45 minutes of their time to participate in the test.

Lastly, I would like to thank Miss Ann Radinovic for her help in reading and keying, correcting, and editing this thesis with the TSO-NSCRIPT system.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Text processing is a term collectively describing the hardware and software systems used to aid in the handling of textual information. Word processing is also used to describe some of these activities, but, usually is more narrow in scope. The Word Processing Institute defines word processing as "the application of modern computer technology and systems management techniques to typewriter communications (12)." Text processing is currently an $800 million industry and is growing at a rate such that it should top $2 billion by 1975 and $8 billion by 1980 (5). One of the driving forces to the increased used of text processing is the fact that administrative costs are one of the fastest rising costs. Text processing systems help in reducing these costs by: 1) enabling typists to work more productively, 2) enabling professionals to be more efficient in creating the documents, and 3) simplifying storage, in that no bulky hard copy need be filed. In addition to reducing administrative costs, there are many other advantages in using text processing systems. Normal editing involves many cycles of writing, typing, and proofreading, but with text processing systems the document must only be updated. Only the updates need to be proofread since there is no possibility of an error occurring in other portions of the document as there is with retyping. The document tends to be better written due to the ease of corrections, and it tends to be better looking for the same reason. More advantages of text processing will be discussed in chapters 3 and 4.

The largest disadvantage of these systems currently is the relatively high cost. However, the costs of almost all of the components of the systems are going down. Communication devices and lines are rather expensive today, but as there are more users these costs should come down. Storage costs have been coming down at a fairly steady rate due to advances in technology, and there is no reason why the trend should stop. The cost of terminals, especially CRT (cathode ray tube) terminals are also coming down. The cost of large computer systems have shown a square root relationship with the "power" of the system since their inception (Grosch's law), and minicomputers offer low costs on the opposite end of the "power" spectrum. With all of these costs going down, and the cost of labor, including professionals, going up, it is not hard to see that there is a potential for a large market in text processing.

Functionally, a text processing system may be divided into three main categories: input, edit, and output. The input phase is concerned with transforming a document from the conceptualizations of an author onto the medium used by the particular system. There may be intermediate steps such as a longhand draft, a dictaphone recording, or cutting and pasting previous documents, but in the end it is usually entered by an operator of a typewriter keyboard based input device. One immediate advantage of the text processing system over normal typing is that errors may be corrected as they are spotted in the input phase. This is usually accomplished by backspacing over the mistake or by typing a "kill character," such as an @ which will delete the character preceeding it (see figure 1.1). It is also possible to shift into the editing mode from the input mode to correct mistakes which have been noticed some time later after their occurrence.

Once a system is ongoing, much of the cutting and pasting can be accomplished by the system simply by transfering documents, or parts of documents, from medium to medium. The predominant media used today are paper tape, cards, magnetic tape (cassettes, cartridges, or reels), and magnetic disks. In some systems, typed pages themselves can be considered a medium since OCR (optical character recognition) equipment can read the pages, and enter the material into a computer. Usually the input phase does involve keying of some sort, however.

Once the text is stored in the system, it is desirable to be able to change parts of the document. This is the function of the editor in a text processing system. An editing command can be broken into three categories. First, the location of the desired change must be made known to the system. This is commonly done on either a line number basis or a context basis. Usually the line number is a measure of a linear displacement of a line from the top of a file. This method of locating the text is awkward from the standpoint that the most up-to-date previous version must be referred to when making changes, and the line numbers change as new text is added. The line number oriented editors are quite acceptable when working with document in which the lines are structurally meaningful, such as a computer program, but are inferior to context oriented editors when applied to other documents. The context method of locating text is more natural since there is no artificial link between the text to be located and the method of locating it. Sometimes more effort is needed with this system since the character string the user wishes to locate may not be unique, for instance, "and the" is a character string which would occur many times in most documents. This drawback can usually be overcome very simply by including a character or two from bordering words. Probably the best method of locating text is accomplished when a television screen--like CRT (cathode ray tube) is used for the terminal. With this system, a large block of text is displayed on the screen and the location of the change is identified by merely pointing to it with an electronic device. Whatever the method, locating is the first function to be done by an editor.

The second step is the specification of the type of editing function to be done. These include substitute, insert, delete, move, or display the desired text. With the delete, move, or display functions, no other information must be supplied other than the locating information. However, with the substitute and insert functions, the third step, textual input, must also be provided. This is usually accomplished by including the new text in the editing command but separated from the other information by the particular systems' separating character, usually referred to as a delimiter. With the use of one of these operations, the text which is stored in the system is changed.

The output phase is the last step in the process, and is responsible for the formatting and printing of the document once it is in its final form. The formatter is responsible for arranging the document into its hard copy form from its input form. This is usually accomplished by a set of formatting commands which can be included in the text itself or reside in a separate file. With these commands, a user may specify the line lengths, margins, centering, etc., at any time during the production of a document, and change them when necessary with little extra work. Better typing efficiency is realized during the input phase because little thought has to be given to formatting by the operator.

Once the format is decided, the document may be produced by a variety of output devices. With most of the systems, the output can be typed on a typewriter-like terminal, but the speed is limited by the mechanical speed of the typewriter. With a few of the systems, the output can be printed on a line printer which is considerably faster than a typewriter. The output can be printed on normal computer paper for drafts or on special perforated paper for final copies. Another approach to output production is to provide input (usually magnetic or paper tape) to photo composition devices. This is especially desirable to the publishing industry or other firms requiring a high quality product. If the output of the text processing system is for information only, i.e. no hard copy is required, a CRT can be used for this display. Whatever the method, the output from the system is likely to be neater and more mistake-free than a document typed in the conventional manner.

It is the purpose of this thesis to: (1) explore the different approaches taken in designing these systems and provide a framework for comparison of each, (2) discuss the common uses and user applications of these systems to date, (3) discuss some advanced and future uses of these systems, and (4) determine the productivity increase due to text processing systems. The systems discussed do not represent a comprehensive list of all that is available, but rather, they represent an attempt to cover the range. Likewise, the applications are not limited by the ones discussed; the imaginative user could probably find dozens more.

FIGURE 1.1
SAMPLE TERMINAL SESSION

```
READY
edit sample nonum new text
INPUT
.ss  (single-space)
.ce  (center)
INTRODUCTION
.sp  (space)
     Although computers have been used to grae@@eat
advantage in mnay areas of the business community
for a number of years, until recintly the field
of text processing has been overlooked.  The
method of turning
out textual infr@ormation
such as reports, contracts, and manuals, has remained
the same while new techniques were
being applied to accounting, management science,
and engineering applications.

EDIT
up 111
TOP OF DATA SET
verify
down 5
advantage in mnay areas of the business community
change /mna/man/
advantage in many areas of the business community
down
for a number of years, until recintly the field
change /recin/recen/
for a number of years, until recently the field
down 3
out textual information
change /ation/ation,/
out textual information,
save
SAVED
end
READY
```

FIGURE 1.1 (cont.)

The output would be:

```
READY
nscript sample
LOAD PAPER; HIT RETURN
```

INTRODUCTION

Although computers have been used to great advantage in many areas of the business comunity for a number of years, until recently the field of text processing has been overlooked. The method of turning out textual information, such as reports, contracts, and manuals, has remained the same while new techniques were being applied to accounting, management science, and engineering applications.

CHAPTER 2

TEXT PROCESSING SYSTEMS

There are many different approaches and philosophies exhibited in the design of text processing systems that are available on the market today. There is a large range of capabilities and, of course, a large range in price--the two usually being directly related. The systems can logically be grouped into three categories: (1) stand alone typewriter terminals, (2) dedicated minicomputers (used for text processing only), and (3) software programs implemented on a multipurpose computer. The latter may refer to a user's own computer or a time-shared computer. A discussion of these three categories and selected members follows.

2.1 AUTOMATIC TYPEWRITERS

These editing systems are designed as "stand alone" and thus do not require a centralized computer, although in some cases, communications with computers is possible. Typically the prices are relatively low, but then so are the capabilities. For instance, extensive editing changes are not easily done, so that with any document of more than fifteen per cent editing changes, retyping may be necessary (5). Their primary selling point is a method of automating the more routine and/or time consuming jobs of the normal office typing load. The typing speed of all systems is on the order of fifteen characters per second, due to the mechanical limitation of the selectric typewriter, upon which all systems are based. Appendix 1 contains a complete listing of most of the automatic typewriters on the market of which three are discussed below.

2.1.1 MT/ST

The Magnetic Tape Selectric Typewriter (1),(9),(19) consists of a single Selectric typewriter connected to a control/memory unit which can include either a single tape cartridge, a dual tape cartridge, or a magnetic card (MC/ST). The purchase price can vary from $3,200 to $11,000 depending on the options (read only, read write, number of cassettes, etc.).

The editing capability in the single card or tape systems is very limited being restricted to the substitution of equal length character strings. If the substitution is not equal in length, but nearly so, the relevant portion of the line may be retyped and hence the insertion is made. With the dual tape system, a little more flexibility is available, but to make a substitution of a larger character string for a smaller one, the entire document has to be copied on to the other tape. This transfer speed is limited to 14.8 cps since the process of transferring text from one tape to another only occurs as the text is being typed by the system. The location of the substitution is accomplished by the operator manually stopping the typewriter during the printing. Paragraph sequences can be rearranged (moved) but usually this complicated procedure makes retyping one of them the easier option. The system can be programmed to stop at preselected points to allow form letter changes, special characters, or page endings. It is also possible to store standard paragraphs on one tape and retrieve them as required to "boiler plate" a letter. Error corrections are made by backspacing and typing over mistakes.

Formatting features include right margin adjustment (but not justification) and variable line lengths. The system will drop unnecessary hyphens when retyping using a different line length, but does not add any new hyphenations.

The magnetic cartridges cost $15 per tape and have the capacity for 24,000 characters, or roughly ten single typed pages. If the documents are to be saved on these cartridges for permanent storage, this represents a relatively high cost of $650 per million characters.

Communication to other MT/ST's is possible via telephone lines with a telephone data set. Computer communication can be accomplished either through a conversion to computer compatible tape via a Digidata

Convertor or directly into the computer via an IBM 2495 tape reader. The Digidata costs $4,600, but is not completely satisfactory in that some trouble can occur if editing codes are left on the MT/ST tape. The IBM 2495 costs about $22,000 and while it is more reliable, it also quite a bit more expensive. With both of these devices the transmission of text can only occur from the MT/ST to the computer not vice versa.

In summary the MT/ST is a very rudimentary system but is a quite cost effective in some applications such as form letters and short business letters.

2.1.2 Redactor

The Redactor editing typewriters (12) can be obtained with single or dual magnetic tape cassettes, or single or dual magnetic cards incorporated in the control unit along with the Selectric I/O typewriter. Prices range from $6,700 for the single cassette or single card version to $8,000 and $8,200 for the dual cassette or dual card version respectively. These cards measure about 3.5 inches by 6 inches and are made of solid mylar. They usually only have enough capacity for one page of text. The cassettes look like the ones used in most commercially available cassette tape recorders, and usually have the capacity for twenty or more single spaced pages.

There are three main advantages of cards over cassettes. First, cards requiring no changes need not be recopied when a document of several pages is being assembled, whereas the whole tape must be. Second, training is generally faster since the concept of one page per card is easier to understand than one cassette for many pages and possibly fractions of pages. Third, if several elements of text are on one cassette, the correct document must be searched for, but the correct document may be retrieved immediately with cards since they can be filed by document. However, these cards do not have the memory capacity that the cassettes have (5).

Editing is accomplished by the same method of copying from one card/tape to another, but this system is a little better in that the transfer rate is 700 characters per second compared to 15 cps for the MT/ST. There is also an automatic document assembly feature where standard paragraphs may be fetched from one tape to the document by programmed instruction codes. One additional feature is that the end of the card/tape is indicated by a tone to prevent the typist from wasting keystrokes. The formatting features are much like the MT/ST's but also include automatic width and tab variations, automatic decimal alignment for columnar work, automatic control for single and double spacing, and automatic underlining.

The cassettes have a storage capacity of 60,000 characters, and cost $9. These storage costs would be a little more reasonable at $150 per million characters. The cards have a capacity for 64 lines of 160 characters per line and cost $1 per card. This yields a theoretical cost of $100 per million characters if every line contained 160 characters. The more likely situation, however, is that there would be one page of typed output per card implying that a more realistic estimate would be $200 per million characters.

There are, at present, no communications control devices for linking with other systems, but they will be available in the future.

The Redactor system offers significantly lower storage costs and a few more features than the MT/ST to make some jobs easier. The types of application, however, are basically the same.

2.1.3 Wang 1200

The Wang System 1200 Cassette Typewriter is available in the dual or single cassette version with a communications oriented package optional. Prices start at $7,000 for the single cassette model and $7,400 for the dual cassette version. When available, the communication control device is priced at $2,400.

In most respects this system is similar to the Redactor system. The editing features are about equal and speeds are the same since both systems use the same type of cassette. The formatting features of this system include those of the Redactor systems but also include automatic heading centering, right margin justification, and page numbering.

One of the strong points of the system is its potential to communicate with other terminals and a central computer. This feature is still under development, but when it is operational it will allow communication in a full duplex mode (transmission is possible to and from the computer). In this manner, documents stored on cassettes can become part of a data base of the computer and used in the preparation of larger documents, and information in the computer can be typed at the console in case line printer quality is not sufficient.

2.2 MINICOMPUTERS

These systems are designed for users who need more capability than available with automatic typewriters, but do not have their own computer or one that is suitable for text processing. Actually, these users form the same market toward which the time shared systems are aimed. The relative advantages and disadvantages of each provide the differentiation. As with any of the computer-aided text processing systems, productivity of typists is greatly increased when compared to manual or automatic typewriters. There are two basic reasons for this. First the terminal operator usually is only concerned with the input and editing modes; the output can be produced by a high speed line printer independently. Second, the editing features eliminate a lot of repeat typing and waiting for text to be transferred from one medium to another. Among the advantages of minis are that, generally, no computer experience is required to operate the system, and security, convenience, and control are more readily available since the system resides in the users own establishment. The chief disadvantage is the fact that a large investment may be required if the system is purchased, implying that a user had better be sure that he needs the capabilities he is purchasing. Figure 2.1 is a schematic of how a computer based system is organized. Appendix 2 lists many of the minicomputer text processing and formatting systems available today.

2.2.1 Documate

Documate (3),(4),(24) is a system consisting of from three to fifteen modified IBM Selectric typewriter terminals connected either directly or through telephone lines to a processor which contains the logic and storage devices. The storage is divided into active and inactive, and capacity

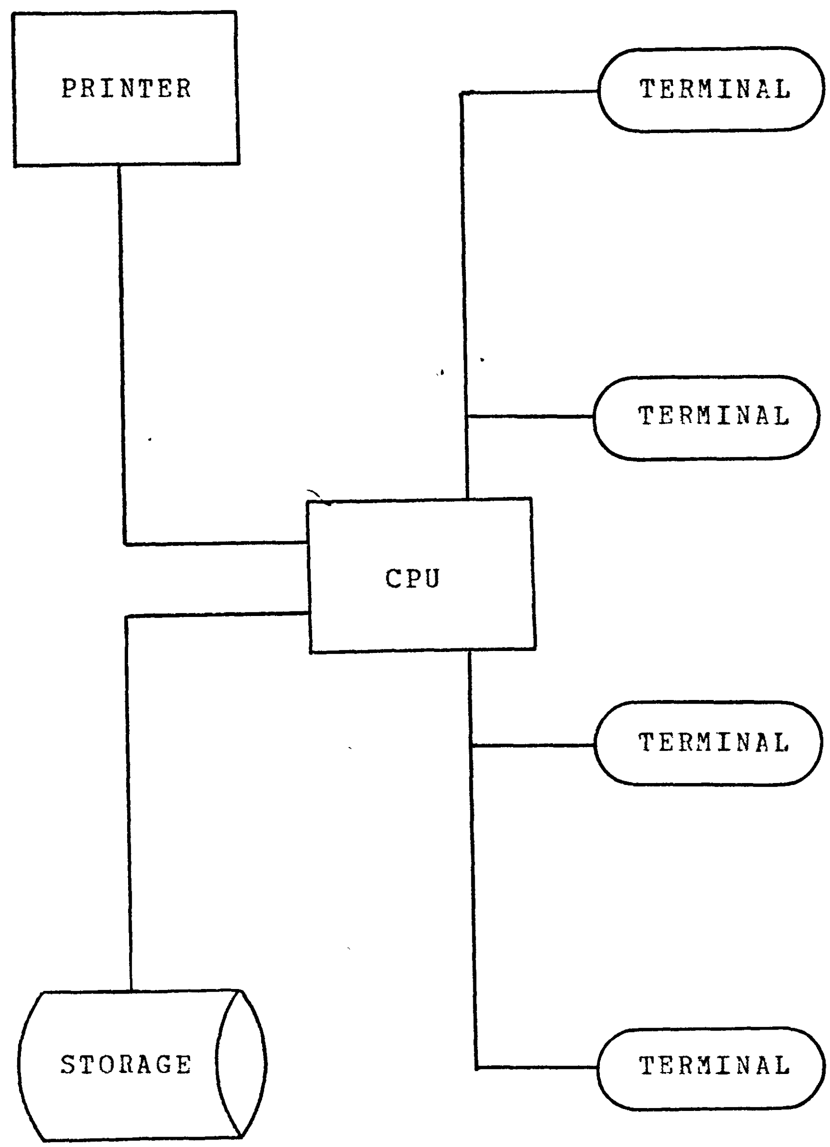

FIGURE 2.1

SCHEMATIC OF COMPUTER BASED TEXT PROCESSING SYSTEM may be added in increments. Also included in the system is a high speed line printer capable of upper and lower case printing. Prices vary from $82,000 to $200,000 with third party leasing and lease-purchasing available.

This system has a fairly comprehensive editing capability. The editor is basically a line oriented system, but it will recognize character strings that overlap lines. In addition, rather than dynamically renumber lines as changes are made, the system adds lines as a subset of the previous line, i.e. the lines following line 52 become 52.1, 52.2, etc., while deleted line numbers disappear. Thus, the line numbers will always correspond to the most recent copy of the document for ease in locating text. A command to renumber the lines in integer sequence may be given when desired. One function of the line numbers is to locate the position where editing is to occur, and commands that will move the pointer up or down any number of lines are available. With Documate the editing position can also be found by specifying context which will uniquely identify the desired location. Ellipses (Now...country) may be used to lessen the typing burden when specifying longer character strings. The searching may be specified to occur in several ranges of the document: within several or all lines preceeding the current position, within several or all lines following, between two specified lines, or throughout the entire document.

Modifications to text include replacing, erasing, changing, and moving any desired length character string. However, when moving large blocks of data, it may be more convenient to label the block as a file and imbed the file in the desired location. This capability is provided by Documate.

The formatting features included are automatic page numbering, right, left, center, or full justification, the four page margins control, headings, footings, and conditional page skipping. It is also possible to set tabs such that columnar work can be left or right justified or aligned on decimal points. All formatting controls are interspersed in the text and must be specified to override any default formatting instructions. It is also possible to keep the formatting instructions in a separate file which is merged with the text when printing the document for cases when the format is variable.

The active storage in the Documate system is provided by magnetic disks which range in capacity from 1.5 million characters to 10 million characters depending on requirements. More capacity can be added when it is needed. The offline storage is also provided by 2315's, but these devices have removable cartridges with 2.5 million characters of capacity, and slightly slower access speeds. The transfer rate of the cartridges to the disk is 5,000 characters per second so that it is not terribly inconvenient to transfer inactive documents back on to the computer. The cost of one cartridge is $155 so that at $62 per million characters this is reasonably inexpensive archival storage.

With so much available storage, there should be a fairly extensive file manager system which Documate has. A complete list of files on disk and cartridge is maintained, and it is possible to retrieve, delete, rename, and transfer media, of any file under an operator's cognizance at the terminal. In addition, it is possible to transfer cognizance of a file from one operator to another. It should be noted that unless an operator supplies the correct name and password it is impossible to access any file in this system.

Output may be typed out on terminals or the high speed line printer depending on the volume and the desired appearance of the document. The line printer operates at 240 lines per minute in a background mode so that the operators have priority with respect to the processor's time. This permits operators to use the system continuously in the input and edit mode and receive output at the same time. If typed quality is needed, it is possible to use one of the terminals which would print at the considerably slower rate of fifteen characters per second. Also, if still finer quality is required it is possible to provide either magnetic tape or paper tape input to a photo composition system.

2.3 MULTI-PURPOSE COMPUTER SYSTEMS

Many computer based text processing systems are available for the users own computer or from a time shared computer service. The end product is the same but there are many advantages and disadvantages of either method. The in-house version provides the convenience of receiving output when desired, in addition to greater control and greater security than time shared services. Among the drawbacks are the large capital requirements necessary for more equipment such as disks, core memory, transmission control units, terminals, and special printers. Special coordination with the computer center staff must be made so that the correct printer, paper, and ribbon are used when producing text output.

With a commercial time shared service, the chief advantage is that there are minimal investments made by the user. The service bureaus usually have a variety of paper stocks, and even photo offset equipment may be available. Hence, the user can primarily concentrate on the input and edit phase of document preparation. With service bureaus no computer knowledge is required, and each subscriber to the service benefits from advances in technology since it only has to be implemented on one system. Among the disadvantages are the higher operational costs, and the lack of convenience, security, etc. This approach to computer aided text processing is especially suited to smaller businesses and firms using text processing for the first time.

Four of these multipurpose computer based systems are discussed below.

2.3.1 ATS

Administrative Terminal System (1),(2),(16),(18),(19) is a program provided as a standard package by IBM on the 360 or 370 series computers. Written to operate under OS, ATS will run in a multiprogram environment. Typically it resides in 64K of memory and uses about five per cent of the system CPU time even with many terminals in use. If a user were to implement this system on his computer he must consider the cost of the terminals (usually 2741's), extra disk storage, extra core memory, the extra load on the printer(s), upper and lower case chains for printers, and programming time in addition to the CPU time, all of which depend on the scope of the system to be implemented. Usually the system provides file management but some programming may be necessary to provide control over the users. Security and accounting data are provided for by the use of account names, account numbers, and operator numbers.

As editing systems go, ATS's is rather primitive but considering that it was one of the original systems, this is not too surprising. Location of text is done strictly by line number, and the line numbers change dynamically as editing proceeds. Usually operators work from the previous hard copy and edit from the bottom up to be able to consistantly locate the desired line. The basic editing command is a substitution, with deletions and insertions being degenerate forms. Lines may be easily moved around and deletions of one or more lines is relatively easy, but to insert new lines in the middle of the document, the lines must be typed at the end of the document and moved to the desired location.

Text may be input in the "formatted" or "unformatted" mode in the ATS system. Typing input in the unformatted mode results in output exactly as it is typed in. This mode is used for tables or other nonconcatenated text. Use of the formatted mode treats the text as one continuous stream, which will be left and right justified at the time of printout. The formatting commands are stored with the text and include line centering, headings and footings, page numbering and a stop output command where variables may be typed in at the time of printing. Care should be exercised when using this feature in the formatted mode since the right margin of the line may not be justified if the input is larger than the space provided for it.

Output may be typed out at the terminal or printed on a high speed printer depending on the desired appearance of the document. It can also be transmitted to another output medium such as paper tape, punched card, or magnetic tape which can be used as input to other systems such as photo composition. At a cost of roughly $1 per million characters, magnetic tape can also be used as a very inexpensive archival storage medium.

2.3.2 SPECS

Specification writing was one of the early natural uses of computerized text processing, and was implemented by a few engineering and architechtural firms using the IBM program DATA TEXT. This was adequate, but not entirely satisfactory, so a group named Automated Procedures for Engineering Consultants, APEC, engaged a consulting company, Engineering Computer International Inc., to create a program especially for specification writing. Since most of the member firms of APEC had IBM 1130's, this program was specified to be able to run on this system and was to include the logical structure of a specification in the program. SPECS(6),(13),(17) was the resulting special purpose program. In addition to the $1000 APEC membership fee, the cost of the program ranges from $1000 for implementation on an 1130 to $3000 for a IBM system/360 with

OS. At extra cost, the program may be obtained with upper and lower case printing capability, but a 16K core is required on the 1130 as opposed to the standard 8K.

There are many limitations imposed on the system since it was primarily designed for an 8K 1130. The largest drawback is the fact that the program executes in a batch mode which does not allow for very interactive editing. However, many of the features included in the system are aimed at easing the specification writers task, rather than aiding the mechanical preparation of the document. The editing is done on a context basis and uses the article number, which is the lowest element in the specification structure, as the location identifier. The basic commands are change, delete, and insert but any character string referenced must be completely typed out.

A specification is basically an organized series of articles, usually arranged in outline form. For instance, a typical structure may appear like:

	article no.	level
I heading	1	1
text		
A) heading	2	2
1. text	3	3
2. text	4	3
a) heading	5	4
text		
B) heading	6	2
1. text	7	3
II heading	8	1

A block of text would be any logically connected subset of articles such as everything under the A) heading or A) 2. category.

Some of the editing features of special interest to specification writers are inclusion or exclusion of special notes, phrase options, multiple choice blocks, dating of edition changes, and relational options. The special notes are of great use to the writer during production of the specifications, and are automatically eliminated on the final copy. The phrase options permit global changes through the text, but when compared to global change capability in other text editors, suffers from the fact that all candidates for global changes must be specified during conception. The multiple choice blocks provides for the selection of one of several options from a master specification which will automatically specify additional text. By choosing an option, the writer links several blocks of information such that the inclusion (exclusion) of one block from the master file will cause the inclusion (exclusion) of other block(s). Most of the benefits of these editing features depend on the preparation of a comprehensive master specification.

The master specification is the data base from which all project masters will be derived. For instance, it would include the specifications for copper wire, aluminum wire, and steel wire among other things. If copper wire were selected exclusively for one project, then all reference to the other two types would not appear if the author chooses the copper wire options only. In another case, the specification for finishing woodwork might include requirements for staining and painting. Upon choosing one, the other will be eliminated since it would be nonsensical to specify both.

The formatting capabilities include most of those contained in the standard computer based systems, but SPECS is unique in that the formatting commands reside in a different file than the text. Also unique is the automatic structure generation which is available. It was noted that approximately 85 per cent of changes to master files were formatting changes which were usually of the variety of changing article 15.211 to 15.21a or XV.211. So SPECS is organized in four generalized levels which are automatically renumbered when insertions or deletions occur according to the specified scheme in the formatting file.

The original document is written such that only the level of the article need be identified. In the format file the method of labeling these levels is specified by prefixes and suffixes. For instance, level 1 might be identified by the prefix 15. or XV., the suffix ")" or ".", and incremented by integers or letters. Each level can be described in a similar manner so that complete freedom is possible when specifying the format. These features are included because it is not unusual for an engineering firm to be dealing with a dozen or more different architects, each of whom has different format standards.

While the SPECS program cannot be considered a general text processing program it is of some value. It makes it possible for smaller firms to have a valuable aid in specification writing, one of the largest uses of text processing, and yet have the convenience of in-house processing on a modest computer budget.

2.3.3 SCRIPT

SCRIPT(14),(15) is another program produced by IBM, Cambridge Scientific Center, for system 360 with CP/CMS and 370 with VM. Like ATS, it also operates in a multiprogram environment, but it possesses many improvements. Since it has more features, it uses slightly more core and perhaps more CPU time. This program is available through IBM on a purchase basis and also is used by a few time sharing service bureaus in various forms.

SCRIPT can be used with a variety of editors, but usually it is CMS EDIT, which is basically a context editor. The text which is to be changed may be located by specifying a unique character string or a column dependent character string rather than a line number. However, the system does not recognize a character string that happens to reside on two different lines. This implies that a line is still considered a structural element of a file. Also the system only searches for text that occurs following the present position. Once the place is located by the imaginary pointer, operations such as change, delete, and insert may be applied to character strings, lines, or groups of lines following the pointer position. There is no explicit provision to move text from one location in a file to another but this task can be accomplished by creating a temporary file and copying it in the desired location.

The formatting features include right margin justification, page numbering, headings, footing, four margin controls, tab setting and line controls. Also, there is a provision for including macro instructions for frequently used sets of format controls. The formatting controls are included in the text as it is keyed in but do not appear on the output.

The output may be printed on a high speed line printer or at the terminal as with other systems. Output on a line printer may be translated into all uppercase if greater printer efficiency is desired. This mode would be used for rough draft copies. Output compatible with other printing media is also possible.

2.3.4 FRESS

FRESS(19),(20),(21) is a commericially available program developed at Brown University and contains a number of advanced features for the retrieval of information as well as the usual editing and formatting. It was developed from earlier work on Hypertext Editing System(19) which was an experimental system for studying text handling techniques as well as studying hypertext as a new medium. The system requires 256K of core memory under CP/CMS on an IBM system 360. It is available on a time sharing basis from Brown University or may be installed on the customer's own machine. The price is subject to negotiation. The features of the system are more useful for the author working in an interactive mode than for the secretary producing documents. A more basic system without the extensive retrieval characteristics, STRESS, will soon be available.

The editor in FRESS is also a context editor but contains more features than the one used by SCRIPT. The FRESS editor has the capability for the use of ellipses(...) in specifying the desired character string. This feature not only makes the typing easier but eliminates a lot of mistakes since the character string for which the computer searches must be typed in exactly as it appears in the file. Unlike SCRIPT, a character string which is typed on two lines may be recognized by FRESS. This is because FRESS considers a line to be only a subsection of a greater stream of text, usually consisting of 500 characters at a time. Also, the system can search backwards to locate text that precedes the current position. Another convenient feature is the fact that each line is "saved" when the carriage return is used. A revert command is also provided to undo any accidental carriage return or faulty editing on the previous operation. With other systems the standard operating procedure is to file a block of text every ten or fifteen minutes to protect against losing everything if the computer goes down during input.

An advanced feature incorporated in FRESS is the ability to direct access a portion of text by the use of "labels", which are input by the user and act like a table of contents. Also, the use of keywords is possible, and upon referencing a certain keyword, appropriate segments of text are returned by the system. A given keyword may be assigned to several places in the document, and more than one keyword may be assigned to a given place. All of these features have "memories" such that after traveling through the text by these methods, it is possible to easily return to the original place in the document. These features rank FRESS as an integrated text processor which will be discussed in Chapter 4.

The formatting features and output capabilities are much the same as SCRIPT. However, FRESS also supports an

IBM 2260 CRT which can be used for faster and more convenient editing since a large block of text can be displayed at one time.

2.4 COMPARISON

Which system is the best? That depends on the application for which it is to be used. Clearly FRESS, one of the most powerful systems, may be used to generate form letters, but that probably is not the most cost effective match of system and application. First of all, the requirements of the application have to be known, and then the systems compared on how they meet those requirements. One attempt to compare all of the systems would be to rank them in each of the broad categories; input, edit, and output. The trouble with that is that an implicit value judgement must be made for non-readily quantifiable characteristics. For instance, who can say that a formatting system that provides for footnotes, but, does not have the capability to allow either Roman or Arabic numerals for page numbering, is better or worse than a system that provides the converse. This decision is best made by the user who knows his requirements more explicitly.

Another method of comparison would be to assign a plus (+) to a system in each category in which it is superior, a zero (0) where it is about average, a minus (-) where it is inferior, or not available (NA) where the feature is nonexistent. While this system also involves some subjectivity, the segregation into three ranks is easier and less arbitrary than a strict ranking scheme, and would provide about as much information.

The next step is to provide a framework for comparison, which is both useful and comprehensible. Rather than have one table with every feature on one margin, and every system on another, it was decided to approach the problem on two levels. The first level of comparison deals with the overall system characteristics which seem important, and the second level deals with the editing and formatting systems in greater detail. Table 2.1 is the comparison of overall features, and includes working storage, permanent storage, editing, formatting, and modularity, which is a measure of compatibility with other devices. Table 2.2 is a comparison of the editing capabilities and Table 2.3 is comparison of the formatting capabilities.

This comparison still does not answer the question, "Which system is the best?," since this question still depends on the application. What this comparison accomplishes is to provide a framework, which, when coupled with the table of applications requirements, allows a systematic evaluation to determine which would be the best system for a given application. Such a table of applications requirements, Table 3.1, is given in Chapter 3.

TABLE 2.1

COMPARISON OF THE SYSTEMS

SYSTEM	WORK STORAGE	PERM'T STORAGE	FORMATTING	EDITING	MODULARITY
MTST	-	-	-	-	-
REDACTOR	-	0	0	-	-
WANG	-	0	0	-	0
DOCUMATE	+	0	+	+	0
ATS	+	+	+	0	0
SPECS	0	+	+	0	-
SCRIPT	+	+	+	+	+
FRESS	+	+	+	+	+

(Note that the rows and columns in Table 2.1 are interchanged from those in Tables 2.2 and 2.3 to be consistant with Table 3.1 to aid in this evaluation.)

TABLE 2.2

COMPARISON OF THE SYSTEM'S EDITING FEATURES

FUNCTION	MTST	REDAC	WANG	DOCUMATE	ATS	SPECS	SCRIPT	FRESS
LOCATE-LINE	-	-	-	+	0	NA	0	0
-CONTEXT	NA	NA	NA	+	-	0	0	+
MOVE	-	-	-	+	-	0	0	+
INSERT	0	0	0	+	+	+	+	+
DELETE	-	-	-	+	+	0	+	+
CHANGE	-	-	-	+	0	0	0	+
COPY	-	0	0	+	0	-	+	+
FILE MANIP'N	-	0	0	+	0	0	+	+

TABLE 2.3

COMPARISON OF THE SYSTEM'S FORMATTING FEATURES

FUNCTION	MTST	REDAC	WANG	DOCUMATE	ATS	SPECS	SCRIPT	FRESS
LINE CONTROL	-	-	-	+	+	+	+	+
MARGIN "	0	0	0	+	+	+	+	+
HEADINGS	NA	NA	-	+	+	+	+	+
FOOTINGS	NA	NA	NA	+	+	+	+	+
STRUCTURE	NA	NA	NA	-	-	+	0	0
FOOTNOTES	NA	NA	NA	NA	NA	0	+	+
COLUMNS	0	0	0	+	+	-	0	0
FIGURES	NA	NA	NA	0	+	0	+	+
JUSTIFICATION	NA	NA	-	+	+	+	+	+
NUMBERING	NA	NA	-	0	0	0	+	+
TABLE OF CONT	NA	NA	NA	NA	NA	NA	0	+

CHAPTER 3

APPLICATIONS OF TEXT PROCESSING

There are many and varied applications of text processing in the business world. Basically, any document which undergoes constant revision and modification, or whose bulk is repeated often, is a good candidate for text processing. The analogy in data processing is an iterative calculation, or a repetitive operation, both of which provided early impetus to computer development. However, it has not been until recently that text processing in any bulk has been considered seriously.

3.1 USES

The various applications can be characterized by their requirements of the features mentioned in the framework discussed in Chapter 2, namely: working storage, permanent storage, editing, formatting, and modularity. Table 3.1 is a summary of several applications, which, while not exhaustive cover the spectrum of uses adequately. In this table, several of the categories could conceivably cover a wide range of requirements. For instance, a Sears & Roebuck catalog would certainly require different cababilities than one from a small manufacturing firm producing less than 20 products. A brief discussion of the nature of all these

TABLE 3.1

APPLICATIONS REQUIREMENTS

APPLICATION	WORK STORAGE	PERM'T STORAGE	FORMATTING	EDITING	MODULARITY
CORRESPONDENCE	-	0	-	-	-
CONTRACTS	0	+	0	0	+
REPORTS	0	+	-	0	0
LISTINGS	0	-	+	-	0
DIRECTORIES	+	-	+	-	+
PROPOSALS	+	0	+	+	0
MANUALS	+	+	+	+	0
SPECIFICATIONS	+	+	0	+	0
PUBLICATIONS	+	0	+	+	+

documents follows.

3.1.1 Correspondence

In this context, correspondence is comprised of short letters of no more than 4 or 5 pages. Because of length, the storage requirements are minimal, perhaps on the order of ten to twenty thousand characters. There is also no need for extensive editing capabilities since most of the editing would merely be correcting typing errors, and other minor changes. A lot of short correspondence consists of form letters, and if this is the case, then retrieval of standard paragraphs for "boiler plating" would be a necessary feature. The number of formatting features required would also be relatively few since the standard letter is not very complicated in nature. Depending on the filing system used, the cost of archival storage could be an important consideration if the user intends to store very many of the documents on tape or cartridge. Compatibility with other input or output devices is not very important since typed output is acceptable, and most letters are relatively independent of other documents.

3.1.2 Contracts

Any legal, medical, or insurance contracts or generally, any medium length document (7 to 20 pages) which shares common characteristics of a contract, such as "boiler plating," would fall in this category. Storage requirements would be larger, not only for the length of the documents, but for various input files used as support. The editing requirements are slightly more demanding since global editing would be necessary and large blocks of text are likely to be moved. The formatting requirements are likely to be more than minimal since the structure of a contract can get quite intricate. Compatibility with photo composition equipment might be an important feature since the appearance of the contract is important.

3.1.3 Reports

There are many internal as well as external reports which would be good applications of computer aided text processing. A report prepared by several authors is one example. A report on current topics is another especially well suited example since it can contain more up-to date information due to the decrease in preparation time. The existence of different reports on tape can also constitute a data base for larger documents, such as proposals, if much of the same information is presented in each.

The storage requirements for reports, of course, vary with the length. Typically, it would require on the order of one hundred thousand characters which is not too excessive. The editing requirements include the usual deletes, inserts, moves, etc., but should also have the ability to merge and manipulate several files. When reports are prepared by several authors, or are pieced together from several other documents, this feature is important. The formatting needs would include column alignments, headings, footings, and page numbering in addition to the standard page control. Permanent storage requirements would not be excessive since reports tend to be current and are usually superceded in time. There is no great need for output other than the usual typewritten appearance but compatibility with other computers may be desirable.

3.1.4 Listings

Listings such as parts, inventories, or price are typically good applications of text processing due to frequent complete or partial updating that occurs. The storage requirements would probably not exceed twenty thousand characters, but that would depend on the size. The editing features required would rarely be more than insert, delete, or change so that an exotic editing system would not be needed. Formatting features would include tab setting, columnar alignment, headings, page numbering and perhaps footnotes among the requirements. There would not be a great need for permanent storage of every revision in this application.

3.1.5 Directories

Directories are good candidates for text processing for the same reason as listings, frequent updating of one basic document. The storage requirements potentially could be greater, however, with lengthier directories. The editing, formatting, and permanent storage requirements are much the same as those for listings. However, there would probably be little need for compatibility with other input or output devices.

3.1.6 Proposals

Proposals are particularly well suited to text processing for several reasons. Revisions can be made when they are conceived rather than just prior to the final typing. This feature also enhances management of the document since its status, or any portion of it, can be made known to the manager simply by requesting a printout. The lower total preparation time also means that more up-to date information can be included in those documents which are of the pre-proposal nature, representing something of a progress report to the customer. In addition, much of the information can be retrieved from other documents such as internal reports or previous proposals. Of course, the usual benefits of mistake free, attractively formatted documents are also important for many proposals.

The storage requirements for most proposals would be quite extensive. About one to two million characters worth of storage would be needed for the document itself and the supporting documents. The power of the editing function should also be extensive since proposals go through many revisions before the final product is approved. The system should have the capability to manipulate files since there would be need to merge information from several files. The formatting requirements would probably be considerable due to the structural nature of proposals and due to the importance of appearance. Permanent storage requirements would be extensive but since only the final copy must be kept for any length of time, these requirements are not too great. Compatibility with photo composition printing may be desirable but probably is not necessary.

3.1.7 Manuals

Probably the greatest advantage of producing manuals via text processing is reduction of the total elapsed time in the revision cycle. Thus, the impact of update can be felt in the field substantially sooner. Following this, the author's editing time is conserved since he would only be required to edit the revisions not the entire document. Along these same lines, a manual for one customer may be modified for another, consistent with the differences in product, so that the new manual need not be completely rewritten. Manuals may also be written using other documents such as internal reports or specifications to ease the process considerably.

The storage requirements for most manuals would probably be quite large depending on both the manual length, and the number of support documents or previous editions used. The editing features needed would be fairly comprehensive because of the amount of manipulating and changing of text. Formatting would also be quite important since, often, the layout of a manual can determine its ease of use. Communication to photo composition equipment may also be necessary if the manuals are for customer use.

3.1.8 Specifications

Engineering and architectural specifications are one of the best, and to date, most used applications of text processing. One of the reasons for its widespread use is the fact that specification writing is a well structured procedure, yet a very time consuming one involving key personnel. The chief benefit derived from the production of specifications via the computer is the freeing of the specification writer for other more professional activities. Also of importance are the decrease in production time to meet deadlines and a reduction in stenographic personnel requirements. Linked strongly to the successful implementation of specification writing is the concept of the master specification. Once a master is available the bulk of the specification is produced upon supplying a few key variables to the system (Chapter 2).

The storage requirements for specification writing are on the order of one to two million characters to include masters and various versions during the production period. Ideally the editing requirements should be minimal because of the master specification concept, but practically, extensive editing is required both for the maintenance of master files and the customizing of individual jobs. The formatting requirements are extensive and should include automatic level generation and numbering in addition to other usual features. Since a contract specification is a legal document, the permanent storage requirements are large and are a serious consideration. Capability for photo composition, again, may be desirable, but not necessary.

3.1.9 Publications

Publications can refer to either books, magazines, newspapers, or any other periodical. Text processing can be used at almost any phase of production, from the author's conceptualization to the production of a tape for the printer. In addition to the normal benefits of text processing, a complete record of all revisions made to an article can be kept if the system is large enough. This feature can be used by an author to trace the evolution of his thought on a particular subject, or trace the development of an error in his work. It can also be used to provide important evolutionary information if an article is legally challenged (22).

The editing capabilities, formatting capabilities, and storage requirements would all be quite large if this system were fully implemented. Output compatibility with a photo composition system would also be important.

3.1.10 Other Uses

These have been some examples of typical hardcopy oriented applications which have been used to date. Given a certain volume of work, these can be readily justified economically. However, there are other applications of text processing which involve new concepts rather than new methods of doing traditional tasks. One such concept is sending documents in the form of text files over communication lines. This procedure can replace two traditional methods of communication, mail service and telephone conversations. Of course, not all phone calls or mail will be eliminated by text processing, but there are several examples of each where it can be both more productive and convenient.

With respect to mailing documents by text processing rather than the postal service, there are two predominant advantages, speed and reliability. The delivery time of a document, even if special delivery is specified, can take several days in even the most efficient postal services. As an alternative, a text file containing a document and its formatting specifications can be transmitted over communication lines (usually telephone) to its destination in a matter of minutes or hours. The cost of this service would be more expensive than mail service, but for some documents, the cost of delays would be many orders of magnitude greater than the difference. The document sent by a text processing system also is more reliable than postal service since there is a finite chance for a document to get lost in the mail. However, if the document gets jumbled in the transmission or otherwise is lost, it can merely be retransmitted. This loss can be signaled by the receiver if he is aware of the difficulty, or by the lack of a system verification of a complete transmission. It would be important for both the receiver and sender to have the necessary text processing equipment, in order to reap the full benefits, but service organizations could be used as long as they were sensitive to the necessary speed. It would also be beneficial if a universal text description language (Chapter 4) existed to insure compatibility among different systems.

Text processing can also replace some communications whose speed is on the other end of the spectrum, telephone calls. The telephone provides the most interactive form of communication in the business world, but sometimes the correspondence does not need to be that interactive. In many instances, a question will be asked followed by a long period of research, and finally a reply. This cycle can be repeated in either direction many times during the call, needlessly tieing up the line for hours. What is needed is a method of correspondence that falls somewhere between mail and the telephone in its interactive capability. With text processing, this correspondence can take place in almost the same total elapsed time but it utilizes less resources. The initiator would simply send his questions in the form of a file which would print on the receiver's terminal. When the necessary information has heen gathered, the answers can be sent by phone or by file whichever is more expedient. Another implication of this form of communication is that it allows the person being questioned to decide which matter he should attend to first. With telephone calls, a person is more or less constrained to give his attention to the one who calls him first. With the alternative method, he can see what questions have to be answered as they come in and can use his own judgement as to which item has more priority.

There are also many intercompany information destribution functions that can be accomplished besides the intracompany ones discussed. For instance in the age of Xerox copies, it is not uncommon for a worker to be completely deluged by documents whose contents are of dubious value to him. The subject may be of general interest to him and parts of it may be important enough to him that he would like detailed information. This would apply to every worker and manager to different extents on different parts of the report. A customized report for every one or several versions for different groups would be two solutions to this problem. The first would clearly be too much work, and the second would involve some compromises since there are bound to be cases when a member of one group would want more information than has been allotted to his group. Text processing can aid in this situation on several dimensions. It can greatly facilitate creating different versions of the same document simply by the inclusion/exclusion of different files. For instance, a version of a report which is to be sent to department 1 would contain detailed information of interest to that department, and summaries of other sections of the report. Likewise, versions sent to department 2 might contain a summary of the detailed information given to department 1 and details of other sections. Going one step further, customizing the report for each individual would also be possible with text processing systems. There could be a concept of a recommended distribution list for the various versions. Each receiver would have the option of obtaining more or less detailed information than other members of his group. The level of detail need not be restricted to an on/off option, but may consist of several levels to add more flexibility. This would especially be useful to busy managers who could receive summaries of everything and details only of especially important items. This may involve adopting different writing styles than are used today, but the real intent of writing reports, providing information, would be better accomplished. These new concepts in office communication could radically improve the information flow to enhance the total operation of the company.

3.2 USERS

How much is computer aided text processing used in business? Not nearly as much as it could be, but more than most people are aware of. The reasons that it is not used more are many: ignorance of the full benefits, inertia in resisting doing things differently if it is not absolutely necessary, and misconceptions of the scope of reorganization necessary to sucessfully implement a text processing system. Following are a few examples of text processing users in the Boston area, including how their systems evolved, the system description, and what applications are included.

3.2.1 Stone & Webster

Stone & Webster is a large engineering firm which engages in the design and construction of large electric generating plants for public utilities as well as processing plants for other industries. Its growth is closely tied to the growth of the electric utility companies, which in the past have doubled their generating capacities every ten years. The growth in the company, of course, has been accompanied by a large growth in personnel and paperwork. Compounding this growth in paperwork is the Atomic Energy Commission's requirements for documentation of safety before licensing nuclear plants, and more recently, the Environmental Protection Agency's requirements for documentation of environmental effects. It was becoming impossible to find enough typists to produce the necessary documentation.

To help keep up with this typing load, Stone & Webster began using Data Text, the forerunner of ATS, through a service bureau in October, 1968. Services were obtained first from a bureau located in Washington, D.C. and later, service from one in New York was obtained in addition. As with most service bureaus, there was a capability for the printed output to be typed at the terminals in Boston, but the number of pages required by Stone & Webster necessitated the use of the bureaus' high speed line printers. Consequently the output was printed at the bureau and shipped overnight by bus or plane to Stone & Webster. However, this service was less than completely satisfactory since delays were often experienced due to transportation problems, weather conditions, etc.

Service bureaus usage was discontinued in November, 1970 when ATS was installed on the company's IBM 360/65 computer. This service was centrally controlled by the Office Management department. With the volume of work which was done at that time, implementation on their own computer was the least costly option.

The use of text processing during this period not only allowed the work to be completed, but also was responsible for keeping the growth rate of the stenographic department less than that of the total administrative personnel.

The ATS department, today, consists of 43 operators using IBM 2741 terminals which are located in four buildings. Twelve more terminals are to be added in the near future. These terminals are connected to the company's IBM 370/165 operating under OS/MVT. The memory capacity is 1,500K bytes (soon to be upgraded to three million) of which a 210K partition is dedicated to ATS. With this set up, all 43 terminals can be in operation simultaneously without any "thrashing" (periods of waiting). This is felt to be important since it is the objective of the department to keep all terminals in operation as much as possible.

In addition to the 210K "working storage" there is direct access permanent storage provided by two IBM 3330 disks each having 100 million locations available. This is roughly equivalent to 60,000 single spaced typed pages. For documents which are not currently being worked on there is also magnetic tape available for storage. The only storage problems at the present time occur because users are reluctant to inform the ATS center of the decrease in priority of documents so that they can be transferred to tape when appropriate. This results in too many documents needlessly residing on direct access storage causing occasional saturation. To deal with this problem, a program is being written to automatically transfer non-current documents from disk to tape as more storage is needed.

Output is printed on an IBM 1403 high speed line printer, which is located in the computer center, but is strictly for ATS use. Normally this device prints 1100 lines per minute but with the upper and lower case chain it operates at about 650 lines per minute. Working copies are produced throughout the day and final copies, which require a higher quality ribbon and special font, are produced twice a day.

Although the majority of ATS terminals are in use constantly, the demand on the computer is quite small. With the 360 the CPU was involved with ATS operation about 12 per cent of the time, but with the 370, the CPU is used by ATS less than 5 per cent of the time.

The charging scheme used by Stone & Webster attempts to completely cover all costs incurred by the computer center that result from ATS including terminals, transmission control equipment, printing, storage and CPU usage. These costs are allocated to the department by connect time, permanent storage, archive storage, printing, and tape transfer. The differentiation of permanent and archive storage is to encourage the use of tapes where appropriate. A program keeps track of all ATS usage and a weekly billing report is produced. This weekly report not only identifies who should be billed, but serves as an audit trail in case customers wish to see the details of their "computer bill". To give an indication of the scope of the ATS usage, the annual costs - for hardware alone - are on the order of $500,000.

The greatest single use of ATS is in the preparation of specifications. This application lends itself particularly well because most of the document is composed of standard text, and there are many review levels where changes are likely to be made. In addition to a decreased typing load, reduction of engineers' time and effort in specification writing has resulted from ATS use. Of course, the most important benefit is the reduction of total elapsed calendar time in producing the specifications.

For a period of time, short correspondence was also produced via ATS, but since no significant benefits could be identified, it was discontinued. This short correspondence is now typed with Redactor twin card or tape machines with better results. Seven to ten pages is the usual breakpoint defining long or short documents, but some judgement is exercised by management in determining whether a document is produced by manual, card, or tape driven typewriters, or

ATS. For instance, if a relatively long document requires extensive column layout, then it might be best to type it first, and later key it in to ATS after the columns have been arranged and first revision made. This would transfer the labor intensive portion of production (organization and layout) from a capital intensive machine (the ATS terminal). Another example of typing the first draft of a document would be one where the expected revisions and levels of review are such that the document would probably require complete retyping.

There are two basic methods of originating a document as shown in Figures 3.1 and 3.2. The first method takes advantage of master documents such as standard specification proposals of job lists, which have been created in the past. After retrieval, the master is customized to the particular job and the resulting working document is stored and printed. The second method is to submit an original document to the ATS department which is then keyed into storage and printed. From here, both methods follow the same procedure. The printed document is proofread by an ATS checker and returned to the originator. If changes are required, the originator marks the original which is then keyed in, printed, and returned to him. This process is continued until no further changes are required and the final copy is printed. The document should then be moved from disk to tape. Jobs are handled on a first-in-first-out

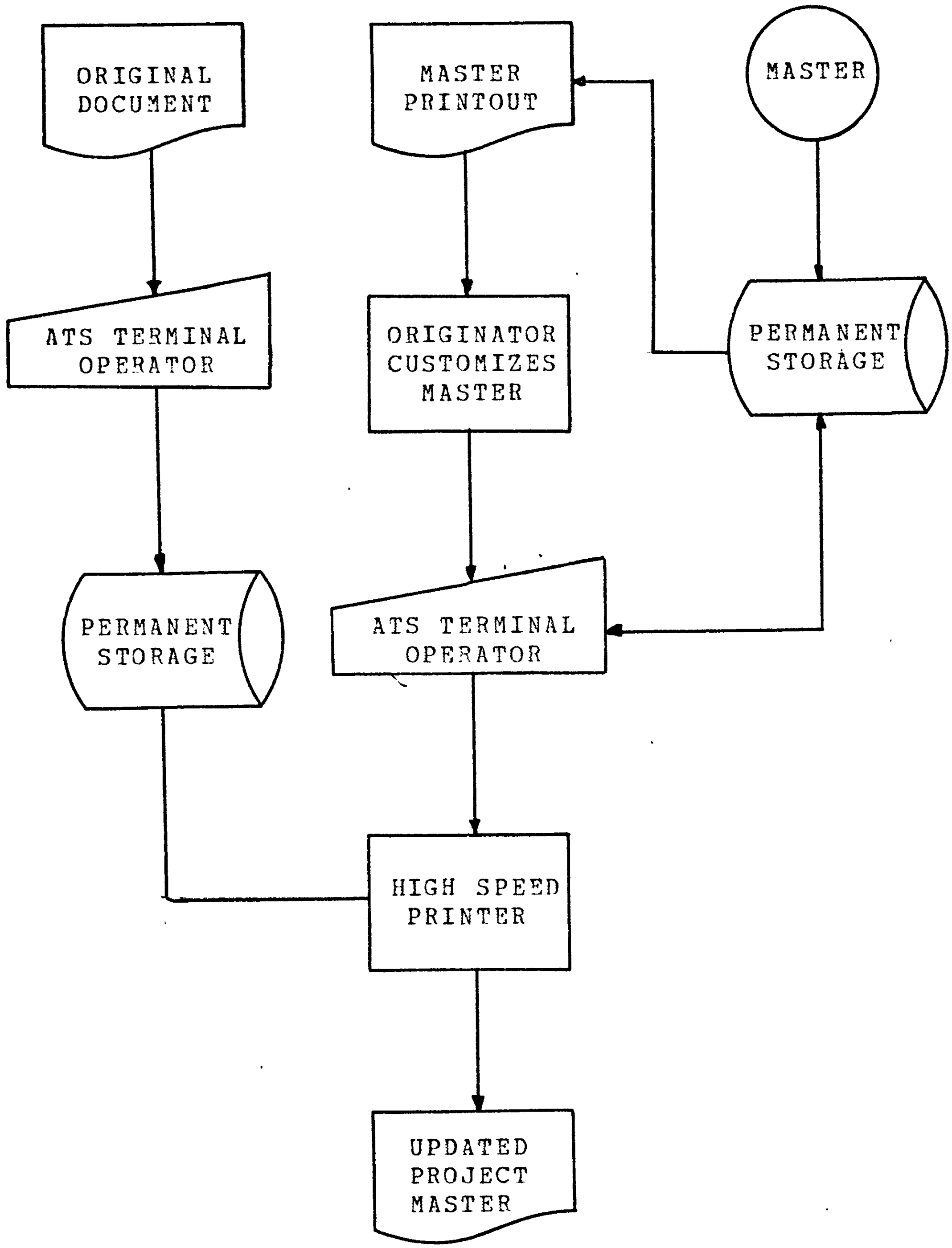

FIGURE 3.1

TWO METHODS OF ORIGINATING DOCUMENTS AT STONE & WEBSTER

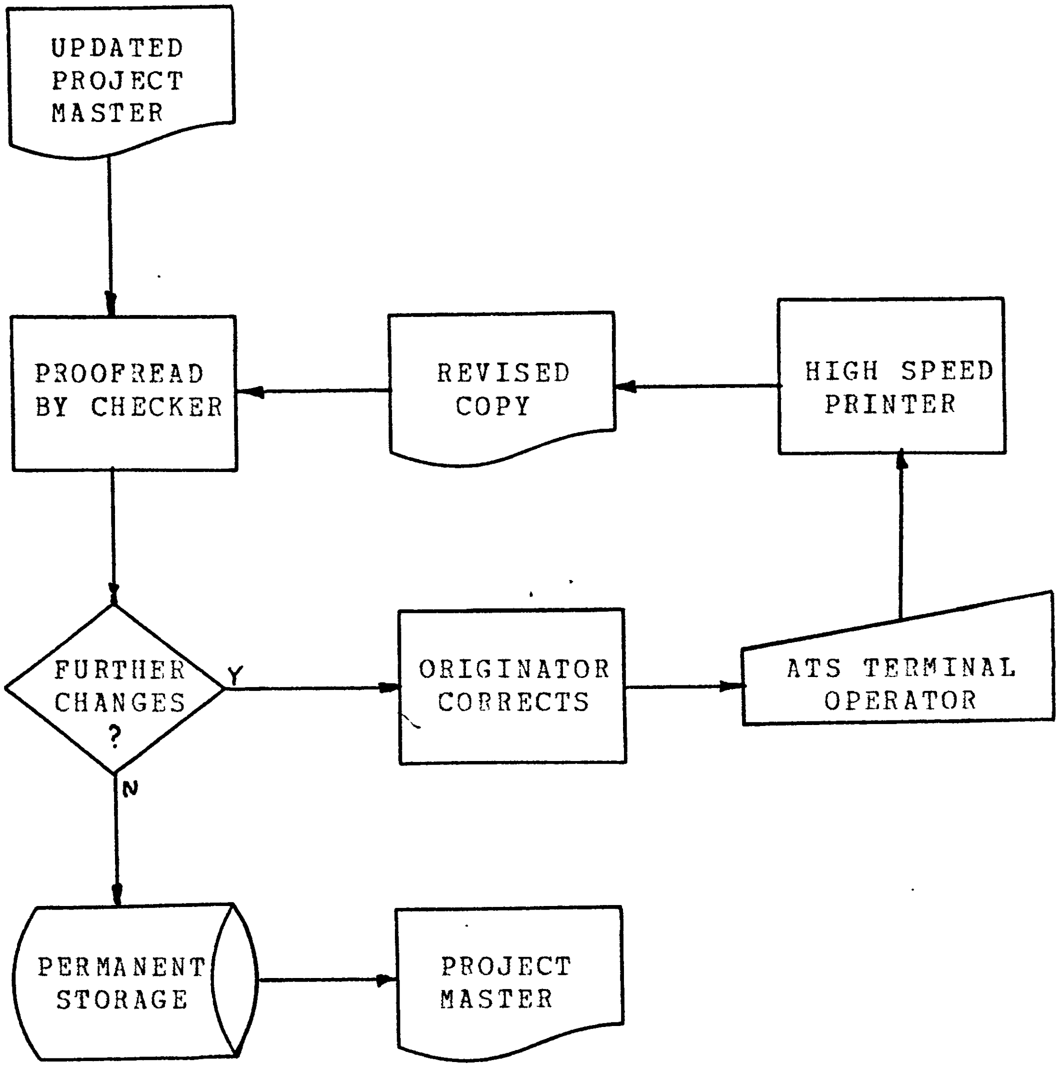

FIGURE 3.2

PRODUCTION OF FINAL
DOCUMENTS AT STONE & WEBSTER

basis, and the typical turnaround, from the time a job leaves an originator until he receives an updated copy, is on the order of three days. While this is not instantaneous response, it is considerably better than the traditional typing methods would ever accomplish. Faster service can be arranged, if necessary, in emergency situations. Some of this delay is due to the centralized nature of the ATS department, since one department's activity, in the form of ATS demands, would be reflected in the service given to other departments. However, for control purposes and efficient operation, it is felt that centralization is necessary.

With computer aided text processing, Stone & Webster has no trouble meeting deadlines because of typing delays. Without the system, they could not even begin to produce the number of documents necessary for their business operation simply because there are not enough typists in the Boston area available to perform these tasks.

3.2.2 Camp, Dresser, and McKee, Inc.

Camp, Dresser and McKee, Inc. is a small to medium sized consulting engineering firm who also produces many specification contracts and also, documents for the Environmental Protection Agency. It is not large enough to require the facilities of an IBM 370 or even a 360. They are a member APEC and like most members they do have an IBM 1130. They typically only operate the computer during the regular eight to five work day, so that some surplus time could be made available for text processing.

Up until the summer of 1971, MT/ST's and Wang 1200 automatic typewriters were the only devices used to do some of the more repetitive chores of the typing load. Time sharing services had been investigated but the one day turnaround of line printer service in addition to other considerations made this alternative less desirable than an in-house system. With all of this in mind, the SPECS program was purchased and installed since it was designed for civil engineering specifications and would operate on the 1130. As a second generation computer, the 1130 does not support multiprogramming, so that card input, rather than terminal, is the most efficient under the circumstances. Secondary storage is provided by three 2315 disk devices, each having 512K words of storage. There is no tape drive implemented on their system so that all storage is on disk or cards. The 2315's have removable cartridges costing between $40 and $90, depending on whether they are purchased new or used. Of the seventy owned by the firm, ten are dedicated to SPECS use. Each word is comprised of 16 bits and may be compacted such that each word contains two characters. This means that a million characters worth of storage will cost between $40 and $90 compared to about $1 for 800 bpi magnetic tape. The only other alternative is punched cards which is an inexpensive method of storage, but not very convenient. A tape drive is available for this system and leases for about $1000 per month. When the requirements for archival storage become such that the acquistion of a tape drive drive wll become more economical, it will be added to the system.

Output is produced on an IBM 1403 line printer which normally operates at 660 lines per minute, but with the upper and lower case chain the theoretical speed is reduced to 270 lines per minute. However, the actual printing speed is more a function of computer processing speed, which in turn is limited by the number of relatively slow disk accesses. Practically, a speed of 50 lines per minute with the normal number of modifications is more realistic.

At CDM, the computer costs are treated as overhead expenses which are allocated to the various departments. The specifications department does not charge customers for computer time since only about 35 per cent of specification work is done by SPECS. Under these circumstances no fair charging scheme could be easily implemented. The amount of computer time available to SPECS is a result of negotiations at budgeting time, and is presently between eight to ten hours a week, with additional time available for emergencies. Since SPECS operates under the master file concept, only jobs for which a suitable master exists or can be constructed, are produced with SPECS. This current limitation to the number of jobs run by SPECS is being improved by a concentrated effort to write and maintain up-to-date master specifications. A second shift on the computer is planned for in the future as more of the work is shifted to SPECS.

The typical job using SPECS is initiated by a specification writer who chooses the various options from the master file, thereby producing a project master file of which he gets a printout. He then makes additions and/or corrections and submits the job to a SPECS trained keypunch operator. She then prepares the deck, submits, and returns the output to the originator when printed. The average turnaround is less than a day, but jobs are only run on specified days due to the special paper and print chain needed. This routine is repeated as the document passes through the required levels of review until it is accepted. The updated project master should be saved for legal reasons until the project is completed.

A primary benefit of SPECS, as identified by the manager of the Specification Writing Department, is the amount of time it saves the writers themselves rather than a reduction in the typing load. In addition to increasing throughput, this savings of time is used by the writers to keep up-to-date in the latest specification requirements by the various regulatory agencies. Another benefit mentioned was the better quality, error-free documents which result. Computer aided text processing is not a necessity for this organization, but is felt to be of great value.

3.2.3 IBM, Cambridge Scientific Center

Not surprisingly, IBM is a large user of text processing. The Cambridge Scientific Center, part of the data processing division, uses the SCRIPT program, among other in-house text processors, implemented on their IBM 360/67 operating under CP-67/CMS. The system is used by programmers, managers, secretaries, or anyone who has any reason to use it. This broad range of access has one side benefit in that there is an improvement in the quality and frequency of programmers documentation. This is no small improvement since engineers and programmers are notorious for their dislike and inability to write. This phenomenon of better documentation occurs in part because it is a system with which the programmers can more readily identify, ie. programming and computers rather than pen and paper. Also, the ease with which revisions and modifications are made encourages improvements to the point that a precision in writing results which is usually not attainable

The SCRIPT program is used as a design tool by at least one programming manager at IBM. In the evolution of the design of a programming system, English, rather than any programming language, is used to completely describe the system; what it is supposed to do and how it is supposed to do it. As new ideas and improved methods are introduced into the system the current edition of the system description is updated by the use of the SCRIPT program. Since many of the contributing system designers are scattered throughout various geographic locations, SCRIPT facilitates and generally improves the communications, since each designer has access to the current edition of the system description via the computer. To prevent unwarranted confusion, the updates are scheduled so that there is an orderly succession of editions. In the intervening periods each designer can keep his own contributions in his own files.

The SCRIPT system is also used in languages descriptions. Typically describing a program language is a very evolutionary process, and again the capabilities of SCRIPT are well suited to the task of maintaining an up-to-date description. Quarterly reports are also prepared using the SCRIPT system and transmitted to New York by telephone line in the form of a SCRIPT file. The reports tend to contain more up-to-date information and, thus, less forecast data, since the document preparation time is reduced. The use of the phone lines instead of the mail for sending the document also adds to this time savings.

These examples have shown a wide variety of uses and also a wide variety of organizational philosophies with respect to text processing. What should be clear at this point is that there is no "best" type of application, or "right" method of organization. It is simply a matter of custom fitting the system to the particular business which, from the full range of systems available, should not prove to be too difficult a task.

CHAPTER 4

INTEGRATED TEXT PROCESSING SYSTEMS

Integrated text processing (7) is an advanced concept which incorporates information retrieval as well as the editing and formatting features of most text processing systems. In addition to these basic characteristics, there should be included in the system: 1) a data management system for handling all stored information, and 2) modular design such that any part of the system can function or be added incrementally by itself. These last characteristics are not really text processing functions, but are aspects of the system which, if included, add to the effectiveness of the system.

The many reports, proposals, specifications, contracts, etc. of a business, in essence, constitute a data base of information needed by a firm for everyday operation. This information can be read into a computer, or stored there as a document as it is produced by one of the many text processing systems. The programs that organize and manipulate these files are collectively referred to as a data management system. For most larger computer systems, these systems are readily available, and are usually part of a computer's operating system. Each block of text (a large document may be broken up into several blocks) can be assigned a file name or number to identify it. The data management system would make it possible for the text processing system to access any of the stored information. Most of the computer based text processing systems have this capability.

The identifiers assigned to each file may be adequate for keeping track of the documents in the data base. However, if the data base is large or is to be kept for a long time, the text processing system must be able to retrieve data by additional identifiers supplying more detailed information. These additional identifiers would include bibliographic information such as author, subject, or title, and index information such as keywords supplied by the author. The user then retrieves information by supplying some logical combination of keywords and bibliographic information. If the number of documents identified as sources is too large or small, the user would use a more restrictive or general combination. Once the desired number of relevant documents is retrieved, they may be made available to the user in a variety of ways. For instance, if a reading list or bibliography is being assembled, the output from the system would be merely be a listing of document titles, authors, and dates. If the user is "cutting and pasting", he may wish a printout of the entire text. In both of these cases, the output may be hardcopy, a visual display on a CRT, or another file in the data base. Only a few text processing systems have these features incorporated in them currently.

The editing system used in an integrated text processing system should be a context editor as opposed to a line editor. Line numbers may also be used to provide for the supplemental locating of information, but there should be no restrictions in manipulating text contained on several lines. The context editor not only facilitates making corrections because it is a more natural way of working, but it permits limited information retrieval since it can locate character strings. This is a useful feature when making global revisions, or making corrections wherever a certain subject matter occurs. Most of the newer computer based text processing systems use context editors.

The formatting commands of an integrated text processing system should be a part of a generalized text description language. Most of the existing systems have the formatting commands interspersed in the text and are usually unique to the particular system. These only describe the desired appearance of the document and nothing of the structure which the appearance attempts to convey. The information structure refers to such things as headings, emphasis, footnotes, captions, or any other predefined type. With this description language, the text would be identified with tags which would specify the structural relationship of that particular part with the rest of the document. The tags would then be translated by a user defined code which would then be interpreted by the output device to produce the desired appearance of the document. For instance, if a word is to be emphasized, the output code for a line printer might specify underlining, and the output code for a printing system might specify boldface type. If a footnote is used, the output code for a line printer might specify merely typing at the bottom of a page, and the output code for a printing system might specify placement at the bottom of the page and a smaller type. The structure may also help in data retrieval since a user may want to look at all of the headings in a document, for instance, to search for some particular piece of information. The SPECS program does use a structure definition to specify format, but is limited in scope. At present there is no universal text description language.

In a text processing system which is fully modular, each program is the smallest possible self-contained unit which performs a single function. Figure 4.1 shows how this modularity should be implemented in an integrated text processing system. The input modules would be written by the user for each of his normal text entry devices, perforated tape, automatic typewriter tape, optical character reader, etc. The translated code would then reside in the data base and be identified as a unique entity by its file name. Information contained in other files is available for this file by the retrieval system, and can be

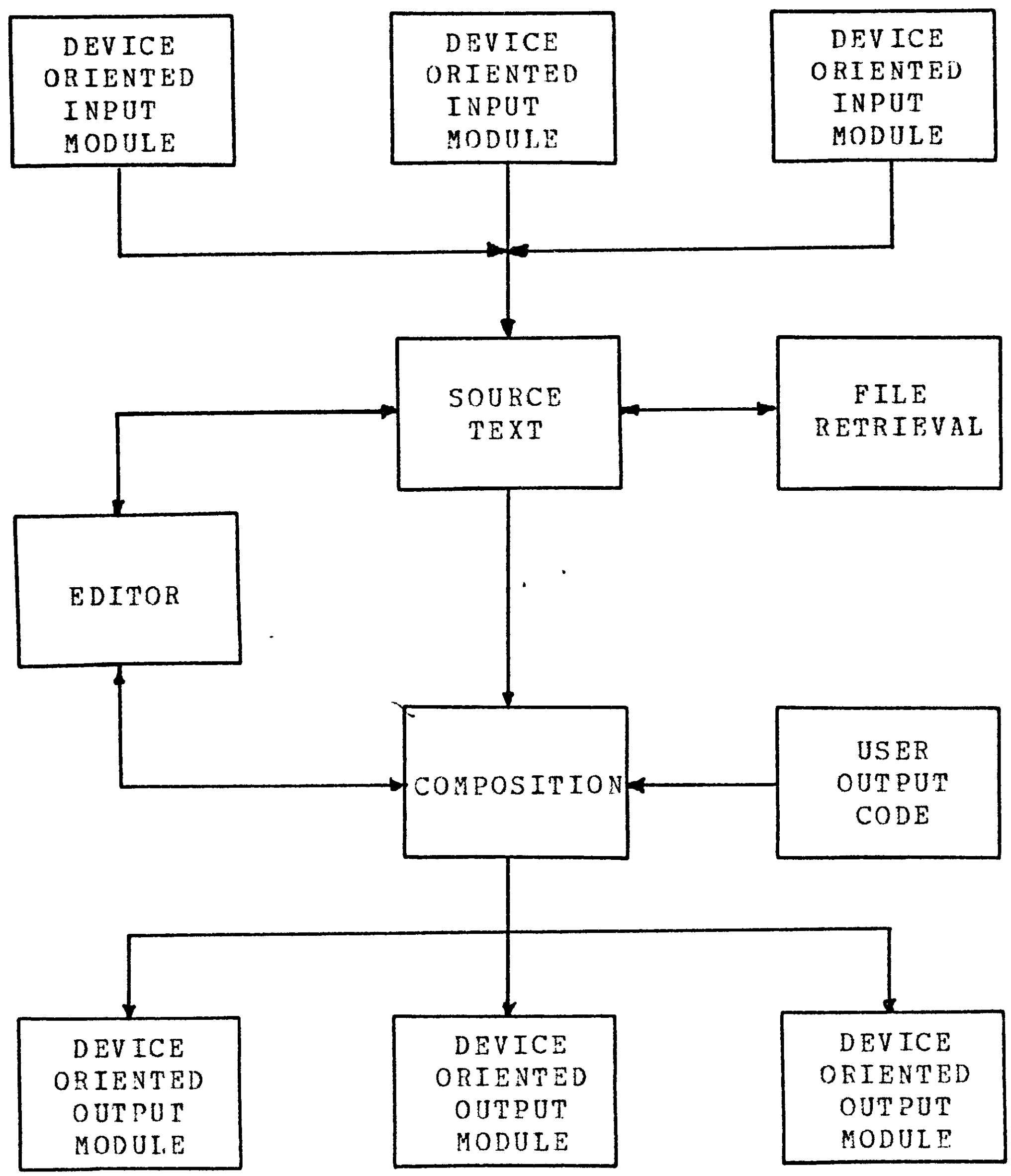

FIGURE 4.1

MODULARITY IN AN INTEGRATED TEXT PROCESSING SYSTEM included in the new file immediately or later during composition. The context editor can be used to operate on the document in any phase of its production; during input or just prior to final composition. The system would refer to user defined output codes, and then produce output files which would include generalized codes for a variety of output devices. At this point, user written output modules would translate the code into device dependent coding for final production.

Modularity permits text processing systems to be customized for the individual users, and tailored to conform to any implementation constraints of particular hardware, operating systems, or data management systems. Also, modularity permits an easy conversion to an integrated text processing system since features can be introduced incrementally as they are needed.

4.1 EXAMPLES OF INTEGRATED TEXT PROCESSORS

While no system possesses all of the features described in the previous section, there are a few who incorporate a significant number of them to be classified as an integrated text processor. The Cambridge Scientific Center has put together a system from existing programs to prove the feasibility of such a system. The system, INteractive Textual Information Management Experiment (INTIME) (7), is operational on their IBM System/360-67, but for IBM internal use only. File management is provided by CMS.

Creation and modification of text is done interactively by the program CMS EDIT. This editor is a context editor and can also address lines by line number and relative position. The document storage and retrieval is accomplished by the IBM System/360 Document Processing System (DPS). The system creates a keyword dictionary from the text and also stores bibliographic data fields. The SCRIPT program provides the formatting from the commands imbedded in the documents. This program lacks the generality for the text description language discussed earlier, but is suitable for line printer output.

The INTIME system is used by the IBM publications staff of a program development group to prepare manuals for publication. These manuals must be revised periodically to reflect changes made to the programs. When such changes are made, the retrieval routines are used to locate the relevant parts of the manuals, which are then revised with the context editor. The final copies are then produced in batch mode by a special user program and a general purpose composition processor.

FRESS can also be considered an integrated text processing system since it contains many of the features. It is described briefly in Chapter 2, and while the retrieval system is not as powerful as DPS, the editing and formatting systems are comparable or better than the ones INTIME uses. There is no modularity in the method that FRESS can be implemented since it incorporates all of the systems described. File management is also provided by CMS.

Probably the most interesting feature of FRESS is its operation using a CRT as a terminal. FRESS is not currently implemented on a CRT but its predecessor, the Hypertext Editing System, is, so that the concept is a proven one. The CRT would have essentially three seperate viewing areas where text would be displayed. At several points in the text there would be cross references which have been placed by the original author. These cross references would be of two types: a branch where a choice is made as to where to proceed, and a link where it is optional to digress to further detailed information such as a footnote, or to continue the main text. For example, if there were a footnoted sentence in a document which was displayed on screen 1, and the editor pointed to the asterisk with the lightpen (an electrical pointer connected to the CRT), the footnote would appear on screen 2. A visual line would link the asterisk to the footnote across the screen. If there were a branch at the end of the footnote, indicating that further information were available from several sources, the source indicated by the user's lightpen would then be shown on screen 3, again with a visable line connecting the information to the branch "menu". FRESS will always be able

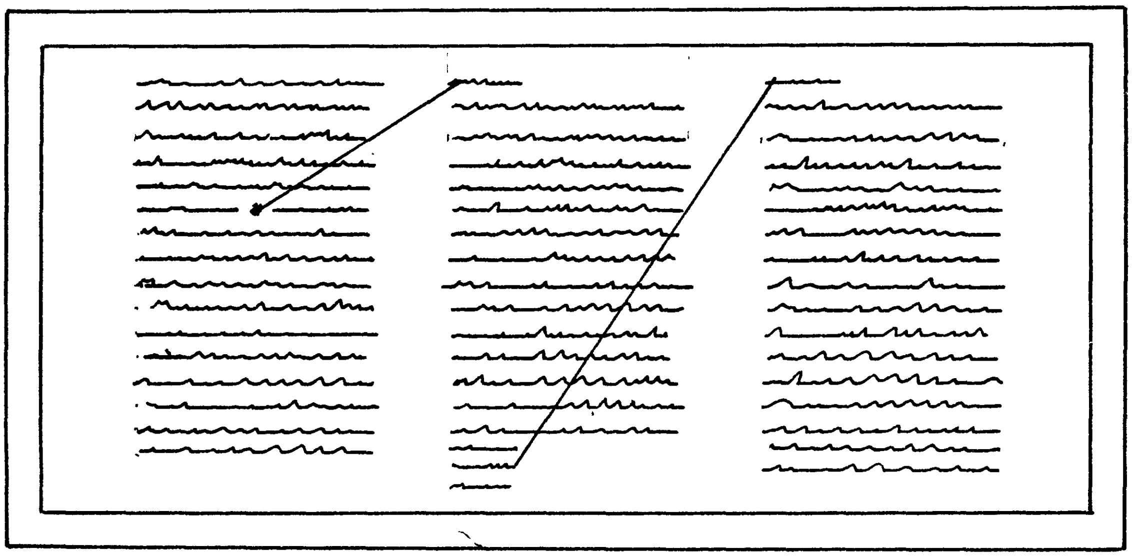

FIGURE 4.2

HYPERTEXT DISPLAYED ON A CRT

to trace the browsing backwards such that the editor will never get lost. This nonlinear travelling through a document is what is meant by the term Hypertext; hyper in the mathematical sense of multidimensionality. This system is aimed primarily for the creation of theses, books, and papers by university personnel, but could have potential for application in the business world.

Another example of integrated text processing is the National Information Center, an information utility on the

ARPA network. All of the networks' documentation, archival storage, and information dissemination is handled here. This is based on the NLS (19) system developed at Stanford University and is implemented on a PDP-10 computer. This system organizes the information in a hierarchial tree structure, as opposed to the continuous string approach of FRESS, with each node containing a statement of less than 3,000 characters. Each statement is supposed to contain a complete thought, but an arbitrary number of substatements can be included for completeness. A user may request to see one section only, all third level subsections of a particular section, the first line of all sections, or any other particular combination to aid him in his information search. There are also elaborate methods of jumping around in the text by means of pattern searches, cross references, statement outline structure, and keyword retrieval.

4.2 USES FOR INTEGRATED TEXT PROCESSORS

There are many potential and actual uses of even the limited scope integrated text processing systems that are available today. One novel use which is implemented at Honeywell is the ability for employees to confidentially seek information on job openings at Honeywell. A good example of the lack of information flow in an organization is the situation where one department or division is laying off people at the same time another is hiring. Also, falling into this category is the situation where a dissatisfied employee looks to other companies for a change rather than his own company because of the various ramifications of letting it be known that he is less than happy with his job. Most firms try to alleviate these situations but the lack of confidentiality tends to put a damper on these attempts. There are a few ways to distribute the information on personnel requirements. One way would be to have a daily list placed on bulletin boards throughout the company, but people would soon catch on if one person was spending an inordinate amount of time reading bulletin boards, and hence, the security criteria would be violated. Another method would be to circulate the list to each employee every day or week, but this would be quite costly if the company were very large. What Honeywell did was to input the personnel requirement list as a file on the computer so that anyone who has access to a computer terminal could see the list completely confidentially. This can be used by managers who realize that they will soon be forced to lay off people as well as dissatisfied employees. Of course, at Honeywell the number of people having access to a terminal is substantial so that it is possible to easily implement the system. However, at other firms who have few terminals, this system would be no better than the bulletin board with respect to confidentiality. As firms use more terminals, systems like this become feasible.

Another use of integrated text processing within a firm which makes use of many terminals, is the concept of on-line manuals. These manuals may be for computer programs, engineering procedures, legal requirements, or any standards which employees refer to in the normal workday. One characteristic of these documents is that new employees or infrequent users need a very detailed presentation for understanding while the frequent users need only summaries. One typical method of dealing with this is to provide both to each employee or group of employees, or only the detailed version. It is usually not possible to provide only the summary since none, except possibly the author is thoroughly familiar with the contents. The result is that a lot of storage space is unnecessarily used and/or users spend unnecessary time leafing through many pages before they find what they want. With integrated text processing, the manual would be stored in the computer and the individual can use them in several ways. If he typically only uses certain sections, he can have these printed as he needs them and has the option to discard or keep them according to his expected future use. In addition to, or instead of, this type of use, the manual could be be input in several levels of detail which would provide complete flexibility for the full range of users. The display of the manual could be by means of a CRT, printed on paper, or some combination of the two.

For instance, it might be convenient for the first or first two levels of detail to be in printed form for quick reference by the user. The next levels of detail could be displayed on a CRT since that type of detail will be needed only infrequently. The fact that the manual is on line and there is only one copy (with backups) means that the process of updating a manual, a nightmare under normal circumstances, can be accomplished quite easily.

One other possible use of an integrated text processing system in businesses is to provide a more complete document retrieval function. Many firms today have some form of computer aided retrieval system, but is usually limited to title, author, date, and a few keywords. In addition to this information it may be desirable to be able to access an abstract as well, to prevent wild goose chases for situations where the author was rather sloppy in his assignment of keywords. It may also be desirable to include a couple of levels of detail in the abstract for even greater flexibility. In time, this type of retrieval system could be implemented in the company library as well to faciitate the information gathering.

Some of these applications are in the future for most companies, but the fact remains that the capability to accomplish them is available today. It will not be too far in the future before these types of activities are commonplace.

CHAPTER 5

PRODUCTIVITY INCREASE

Throughout text processing equipment literature and this thesis there have been many claims of increased productivity in typing and preparation of documents. It is the purpose of this chapter to give an indication of just how much productivity increase can be expected in the production stage of the document. This will be accomplished by relating the first hand experience obtained in the production of this thesis, by quoting results of studies made by existing users in the business community, and by presenting results of tests conducted by the author. Other benefits resulting from increasing the amount of professional time and decreasing total document production time are probably more important, but are harder to measure. It would be up to individual businesses to take advantage of these as they see fit, but it is difficult to put a dollar amount or a dollar translatable measure for objective evaluation. However, if a text processing system can be justified on merely increased productivity then the other benefits would add to the total value.

5.1 THESIS PRODUCTION

The production of this thesis was greatly facilitated by the use of text processing. The system used was one of the ones available at MIT called NSCRIPT, which is a takeoff from the SCRIPT program discussed in Chapter 2, and possesses most of the formatting features required for a thesis. The editing features of this system will be discussed later in this chapter. The motivation for using this system as opposed to typing were: 1) to provide a first hand learning experience on how text processing works, 2) to provide a proof of principle, 3) to provide free labor.

The learning experience was very useful in that it provided a better framework for comparing the various systems. Without first hand experience, the importance of various features and capabilities might be judged according to how impressive the company literature presents their arguments. Also, many of the operational realities of these systems were revealed, so that features designed to eliminate operational problems can be appreciated more. One of the first problems realized was the fact that university computers go down quite often during terminal sessions, which can result in a lot of work being wasted if the text was not saved periodically. Also, the consequences of a lost file are best learned first hand, and quickly lead one to take protective measures such as shorter files with duplicates of each on tape. In addition, some of the organizational problems of document production are manifest by first hand experience. It is soon learned that the various corrections and revisions are best made in an orderly fashion to prevent chaos. Most of these characteristics have been discussed in the literature, but take on a new, and more graphic meaning through experience.

The proof of principle is accomplished by producing the thesis with a text processing system and comparing it with conventional means. A strictly one-off copy, with one level of review, such as a thesis, is not the best application for text processing, but in the author's opinion can still be justified. While most students implicitly allow so much time for final typing and proofreading, this was not necessary with the text processing system, and thus a valuable saving in student time was realized. On a purely economic basis, the two methods are close. At a going rate of $.75 per page for professional typists, the cost of a first draft and a final copy would be about $150. The computer budget used in this thesis was about $300, but some of that was used for the test discussed later in this chapter, and much of it was the result of the trial and error learning process in the early stages. If the thesis were to be done over, the total costs would be less than $150. One major benefit of text processing is the ability to make last minute revisions with little extra effort. While some students forgo these last minute changes, because of the implications of retyping, they probably would include them if they were using a text processing system.

The free labor was provided from a source who is capable of typing about 40 words per minute, but is prone to make errors. Her actual output would then be about three pages per hour after allowance for erasing. Since a thesis must be free of erasures, this would mean many retyped pages in the final form which would considerably lower her efficiency. She flatly refused to type a document of the thesis' length and quality, but was receptive to the idea of using the text processing system. This meant that she would only have to type the basic text once, and make revisions where necessary. Her output increased to more than four pages per hour during the input phase, and was about ten pages per hour in the editing phase, indicating there was a significant increase in productivity on her part. If applied to the production of a first draft and a final copy, this increase in productivity amounts to about 94 per cent. The fact that she would do it with one system but not the other also gives some indication of the morale boosting properties of a text processing system.

5.2 INDUSTRY REPORTS

Not everyone who has installed a text processing system can produce figures documenting their savings. To do this would require the normal "industrial engineered", formalized, work standards program. This includes such things as task analysis, methods-time measurement, and standard average hours, which are difficult to apply effectively to most secretarial activities. There are few organizations that have large typing pools and while some do apply these methods to these typing specialists, others do not for reason of policy. The few published results indicate that significant increases have resulted from relatively simple systems.

One New Jersey office which organizes eight typing specialists into a word processing group, has increased output from 95 lines per day with the old system to 800. On a words per minute basis this means that the secretary who ostensibly types 60 words per minute, actually was producing about three or four words per minute when all corrections and retypings were done. With the automatic equipment and reorganizations, this increased to fifteen to twenty words a minute, again taking into account all the practicalities of setup, referencing and button pushing (10). Part of the increase is due to the reduced pressure of typists, since they no longer had to fear making mistakes which necessitate retyping. This allows typing at draft speeds throughout a document with no slowing at the end when the fear of errors in the greatest. The greatest portion of an increase of this magnitude however, is probably due to the reproductive capability of the system. In another case, a large engineering firm reports that, as a rule of thumb, one computer supported terminal is worth two and a half typewriters for the type of work they do.

The Dartnell Institute of Business Research in Chicago has placed the cost of the average business letter at $3.31 compared to $1.17 twenty years ago. The costs are broken down as follows: "dictator's time (based upon $250 a week salary), 72 cents; secretarial time (base salary, $132), 99 cents; nonproductive labor (illnesses, vacations, etc.), 26 cents; fixed charges (overhead, office maintenance, etc.), 86 cents; materials costs (stationary, envelopes, etc.) 10 cents; mailing costs (including sealing and stamping envelopes), 21 cents; filing costs, 71 cents (24)." In offices where editing typewriters have been used, the costs of the average typewritten document have decreased thirty nine per cent using this formula.(8)

5.3 EXPERIMENT

It is very good that a firm can increase output by 50 per cent, 100 per cent, or even 1000 percent. However, if the documents consist exclusively of form letters where only a name has to be inserted manually, then perhaps a little more breakdown in information is needed on how the increases in productivity were accomplished for the numbers to be meaningful. The capability to copy or modify standard text is a strong function of a particular business, so that benefits due to this feature will also depend on the particular business. However, increases in typing speed for a given time slice can be measured fairly easily, and implications of the productivity increase would be universally applicable. Although the level of this typing speed has little meaning when translated to overall productivity, the change in typing speed would have some meaning for change in productivity. With this in mind, a test was given to ten subjects to identify increases in productivity achievable with the text processing system available at MIT.

5.3.1 The System

The system used was NSCRIPT, coupled with the TSO (Time Sharing Option) editor. Although the actual features of the NSCRIPT system were not used since no formatted output was obtained, the text was input as though it was an NSCRIPT dataset. The TSO editor, when operating on an NSCRIPT dataset, i.e. no line numbers, is a context editor similar to the CMS EDITOR. Text can be located by moving through the document a specified number of lines by means of the UP n, or the DOWN n commands. The FIND command can also locate the desired line by specifying a character string that will uniquely identify the line. This command will only search for the occurance of the character string in the text that follows the current position, and will find the first occurance of the string. Some gymnastics are necessary to find subsequent occurances of the string since the system will not go beyond the first when additional FIND commands are specified. Usually the FIND command is used to locate text which is relatively distant from the current position, and the UP or DOWN commands are used when the number of lines to be moved can be easily determined. The FIND command should not be used exclusively since it uses more computer facilities (CPU) and it usually takes longer for the system to respond and, in the case of the TSO editor, it is more prone to errors.

The three basic commands are the CHANGE, DELETE, and INSERT commands. The CHANGE command consists of the specification of the operator, c or change, followed by a delimiter, usually a slash(/), the old character string, another delimiter, and the new string. A null character string may be used as the new string to effectivly delete the old string. The normal command would appear as such:

```
c/old word/new word/     (carriage return)
```

If the verify mode is used, the new version of the line will be displayed.

The DELETE command is to be used to delete a whole line and would appear like:

d (carriage return)

upon which the line at which the pointer is currently positioned will be deleted. If the verify mode is used, the previous line will be displayed.

The INSERT command is used to insert text immediately following the current line. The typical example would look like:

in new text (carriage return)

If the VERIFY mode is used, the system will not respond with anything at this point.

There are other commands which are involved with various file manipulations in the system, but will not be discussed here.

5.3.2 Subjects

An attempt was made to include people with diverse backgrounds in typing experience as well as computer experience as subjects. Since the system can be used by both typists and non-typists and provide significant benefits in either case, the subjects were recruited in two separate groups. The first group was considered, or considered themselves, to be typists. This characterization usually referred to the capability to use both hands and not have to look at the keys in order to type. The nontypists considered themselves as such and usually were right. They ranged from advocates of the "hunt and peck" method to fairly dextrous people who had to use their eyes to find keys occassionally. Throughout both groups there exists a range of computer knowledge from analyst/programmers to people who never used computers before. There was not a very broad range in IQ since most of the subjects were graduate students or people who seemed to be above average in intelligence. Table 5.1 is a brief description of the subjects, summarizing their typing skill, computer knowledge, education, age, and sex. Numbers 1 through 5 were considered typists, and numbers 6 through 10 were considered non-typists. The typing skill indicator was assigned after the test, since there was some significant variation in the subject's assessment of their typing ability.

5.3.3 The Test

The test was divided into two parts, each designed to obtain information on different aspects of the system. The first part was simply a timed test where the subjects were

TABLE 5.1

DESCRIPTION OF SUBJECTS

SUBJECT	TYPING SKILL	COMPUTER KNOWLEDGE	EDUCATION	AGE	SEX
1	0	+	HS	20-25	F
2	0	+	G	25-30	M
3	+	-	HS	20-25	F
4	+	-	HS	20-25	F
5	0	0	G	25-30	M
6	0	+	G	20-25	M
7	-	0	7th	10-15	M
8	-	-	G	25-30	M
9	0	-	G	25-30	M
10	-	-	G	25-30	M

asked to type a paragraph, first with a typewriter, and then with a terminal which has the ability to correct mistakes, as they are made, by backspacing and retyping. To keep the results as consistent as possible, the two tests were conducted on the same machine, with the terminal in the local mode for the typewriter tests and in the communication mode for the text processing tests. The characteristics of the terminal in the communication mode were basically the same as any keyboard based text processing system operating in the input mode so that the results could apply to any system. To keep the time of the test reasonable, the non-typists were given a reasonably short paragraph to type, of about fifty words, while the typists were given a paragraph of about 150 words.

The subjects were timed on the typewriter for their paragraph and warned that each mistake would result in a thirty second addition to their time to simulate erasing. They were advised to continue typing whenever they made a mistake, however. When typing on the text processing terminal, the same thirty second penalty would be assessed for mistakes, but the subjects were encouraged to backspace over mistakes and correct them when they were noticed. This thirty second penalty could be attributable to the various editing operations that would have to be made, if the mistakes were to be corrected later with the text processing system. This is a little exaggerated for mistakes that would be corrected by the editor during the input phase, or several corrections in one session, but would be reasonable for the corrections of one or two mistakes at a later editing session (Chapter 1).

The second part of the test was designed to obtain some information on how easy it is to learn to use the editor and how effective its use is initially. Each subject was given about a ten minute lecture on the three basic editing operations of the TSO editor, CHANGE, DELETE, and INSERT. Also included were instructions on how to locate the desired line by means of the FIND command or the UP and DOWN commands. The use of the VERIFY mode was encouraged so that they could have some confidence of the correctness of the changes that were made. After the lecture, during which the operations were explained at least twice, the subjects were provided with a summary of the commands for reference during the test.

The test was basically the same for each subject in the type of changes to be made if not the same exact changes. The first change consisted of deleting a string of characters which made up twenty per cent of one line and eighty per cent of the following line. The next change was a simple substitution of one word for another, and the last change was an insertion of a sentence somewhere in the text. The first change could have been accomplished by either three steps (a substitution of a null character string, a deletion, and an insertion), or two (a substitution and a deletion). The second change could only be accomplished in one way, and the last could be done in one of two ways, neither of which was any better than the other.

Some prompting was given during this test if the subject was clearly baffled, or was needlessly waiting for a system response, or was not sure of the exact format of an editing command. The subjects were measured for the time it took to successfully make the changes. If mistakes were made during the editing steps they were expected to correct them using the same methods. After the first timed editing test, the basic commands were reviewed, hints on how to more easily accomplish the same changes were given, and mistakes were explained. Three similar changes were made to the current edition of their text, and a second timed test was given, this time with little or no prompting.

5.3.4 The Results

The results of the timed tests appear in Table 5.2. Following subject 1, the results read like this. The test on the typewriter yielded 39 words per minute without error corrections. Since five errors were made, the net rate is reduced to 23 words per minute after adding the penalty seconds to the time. With the terminal, the gross rate was 33 words per minute and six errors were made. However, five of these were corrected during the test by backspacing and retyping, so that the net rate was 30 words per minute. The difference in net words per minute between the two tests was a 30 per cent increased with the terminal. It can be seen that almost all of the subjects increase their net rates (58 per cent average) with the terminal, indicating that significant increases in typing productivity can result with text processing systems.

One of the first phenomenon observable is the fact that almost all of the typists decreased their gross words per minute when they switched from the typewriter to the terminal. There are two opposing forces that would change the raw speed of the typists during this switch. First, the capability to fix mistakes easily would permit the typist to type more rapidly, and second, the actual time consumed in

TABLE 5.2

RESULTS OF TIMED TEST FOR
TYPING AND TERMINAL INPUT

SUBJECT	TYPED WPM	ERRORS	COR'D WPM	TER'L WPM	ERRORS (NET)	COR'D WPM	CHANGE CWPM %
1	39	5	23	33	6(1)	30	30
2	34	16	12	22	21(2)	19	58
3	60	3	36	52	3(1)	45	25
4	57	11	18	48	5(0)	48	167
5	26	4	19	27	7(4)	20	5
6	21	4	11	24	3(0)	24	118
7	7.7	20	2.9	8.6	8(2)	7.2	148
8	8.1	2	6.9	7.8	1(0)	7.8	13
9	32	0	32	37	0(0)	37	16
10	10	0	10	11.5	2(1)	10	0

TABLE 5.3

STATISTICAL SUMMARY OF TIMED TESTS

SUBJECT	TYPED WPM	ERRORS	COR'D WPM	TER'L WPM	ERRORS (NET)	COR'D WPM	CHANGE CWPM %
TYPISTS							
MEAN	43.2	7.8	21.6	36.4	8.4(1.6)	32.4	57
STD ERR	6.6	2.5	4.0	5.8	3.4(0.7)	6.1	28.7
VAR	14.7	5.5	9.0	13.0	7.6(1.5)	13.6	64.3
t							1.98
PROB > 0							.940
NON-TYPISTS							
MEAN	15.8	5.2	12.6	17.8	2.8(1.6)	17.6	60.2
STD ERR	4.7	3.8	5.0	5.6	1.4(0.5)	5.8	30.6
VAR	10.6	8.4	11.3	12.5	3.1(1.0)	13.0	68.5
t							1.97
PROB > 0							.939
TOTAL							
MEAN	29.5	6.5	17.1	27	5.6(1.1)	25.0	58.0
STD ERR	5.8	2.2	3.4	5.2	1,9(0.4)	4.7	19.7
VAR	18.4	6.9	10.7	16.6	6.0(1.3)	14.8	62.4
t							2.95
PROB > 0							.992

identifying and backspacing over a mistake would add time to the test. The latter seems to be the stronger in this case, since the decrease in speed seems to vary directly with the number of corrected mistakes.

On the other hand, almost all of the non-typists increased their typing speed when they switched. Since these people usually looked at the keys as they typed, they tended to make fewer mistakes and hence would not take any additional time to correct mistakes. A plausible explanation of the increase would come from the fact that it was the second time that they had typed the relatively short paragraph and a learning phenomenon occured.

The errors committed were about the same on average for the typists, but the ability to correct them was very significant since about eighty per cent of the mistakes were eliminated during the terminal session. This error correction ability accounted for an increase in net words per minute for all the typists except one who did not spot most of his mistakes. In the case of the non-typists, the number of mistakes made were about the same in both trials except for one relatively spastic seventh grader who learned something about key position by his second attempt. Again, about eighty per cent of the mistakes were corrected at the terminal allowing an increase in net words per minute for this reason, in three of the five cases.

Table 5.3 is a statistical summary of the results reported in Table 5.2. Included are the mean, the standard error of the mean, and the variance, for all of the columns in Table 5.2. In addition, t-statistics are presented for the increase in net words per minute column. As would probably be expected from a sample this small and diverse, the variance of all the categories is quite large. Combining the two groups only makes the variance worse for the speed measurements since the groups are significantly different in their typing capabilities. However, the question to be answered is "are the increases in productivity significant, or are they due to chance?" When one standard deviation of the average net words per minute done on the typewriter is on the order of the increase in average words per minute, it does not seem if the productivity increase can be inferred with very much confidence. This assumes that the speeds are distributed normally, but problems can arise when samples of this size are assumed normal. The t-distribution differs from the normal distribution primarily for small degrees of freedom (small sample sizes) and hence, is better suited for this application.

The null hypothesis in this case is that no increase in productivity will result in the input phase of text processing systems vs. normal typing. This implies that the mean of the change in net words per minute from one system to the other is in fact zero. The value of "t" in this situation is the mean divided by the standard error of the mean (the variance divided by the square root of the number in the sample). Using this procedure, the value of "t" for the typists is 1.98 (4 d.f.), 1.97 (4 d.f.) for the non-typists, and 2.95 (9 d.f.) for the total. From tables of t-statistics, these values imply that there is a 94 per cent chance that the mean is greater than zero for the two groups taken separately. With the combination of the two groups, the probability that the mean is greater than zero climbs to 99.2 per cent. With these two facts, it can be concluded that the null hypothesis is rejected, implying that increases in productivity will result in the input phase of a text processing system over typing. To give added credibility to the individual increases, subject number 1's increase of 30 per cent compares favorably to her increase in productivity from three pages per hour to four pages per hour in the production of this thesis.

The results of the editing test appear in Table 5.4 and the statistics are summarized in Table 5.5. The purpose of this test was to get an idea of how easy it is to teach someone to use the basic editing commands of a text processing system, and how effective they can be with the system initially. Two tests were run on each subject to get some idea of how the learning curve proceeds initially. There have been some tests conducted by firms indicating that terminal operators reach the 50 per cent performance

TABLE 5.4
RESULTS OF EDITING TESTS

SUBJECT	TIME 1 (SEC)	TIME 2 (SEC)	CHANGE (SEC)	% CHANGE
1	378	179	199	53
2	329	220	109	33
3	379	267	112	30
4	309	265	44	14
5	215	162	53	31
6	118	65	53	45
7	302	*	*	*
8	471	270	201	42
9	598	371	227	39
10	392	213	179	46

TABLE 5.5

STATISTICAL SUMMARY OF EDITING TESTS

	TIME 1 (SEC)	TIME 2 (SEC)	CHANGE (SEC)	% CHANGE
TYPISTS				
MEAN	322	219	103	32.2
STD ERR	30	21	28	6.2
VAR	67	48	62	13.9
NON-TYPISTS				
MEAN	376	230	165	43.0
STD ERR	81	64	40	1.6
VAR	181	128	79	3.2
TOTAL				
MEAN	349	224	131	37.0
STD ERR	42	28	24	3.8
VAR	131	85	72	11.4

average after two months service. They reach the 75 per cent performance averave after four months, and attain an 80 per cent figure at the end of six months (10). This would seem to indicate that some penalties must be paid during the startup period. However, this test shows that a person can be quite effective initially, and increase his/her performance by 37 per cent after the first trial.

The time it took for each of the subjects to accomplish the three changes is recorded in the first two columns of Table 5.4. The changes in the second trial were similar in nature to the ones in the first trial, but were not exactly the same. Most took about 3-400 seconds, and the lowest time can be explained by the fact that the subject used system editors before in conjunction with programming work. The longest time can be explained by the subject's refusal to learn any other command than the c/..../.../, with which he made every change. This not only took longer to type, but was more prone to errors. It generally took the non-typists longer to make the changes than the typists as the means on Table 5.5 show, but this is probably due as much to differences in typing speed as anything else. The average decrease in time required to make the three changes of the second trial was about thirty per cent for the typists and about forty per cent for the non-typists. Both the decrease in time and the per cent decrease in time are significantly different from zero by t-statistics.

Therefore, learning actually proceeds quite rapidly when learning this system initially.

If the average time to make corrections is combined with the average typing speeds (net) for the typist group, then it can be inferred that for any document longer than 117 words in length, the text processing method would be more efficient to make corrections to a document than retyping, even if the terminal operators are rookies. The break-even length decreases to 79 in the second trial, and if the most experienced operator's time is combined with a 60 words per minute rate, simulating an accomplished operator, the break even length for correcting these three mistakes is 65 words, or about five lines. This would imply that there can be as much as sixty per cent changes in a document (number of corrections divided by the number lines) before retyping is the more attractive alternative once the system is fully implemented.

Thus, it has been shown that productivity increases will occur due to the ability to correct mistakes during input, and the ability to make revisions to part of a document instead of retyping. All systems have the capability to correct errors as they are spotted during input, but not all systems allow editing changes with equal ease. The greatest increases in productivity would potentially occur because of the system's editing capability, but these features are the most costly. It has also been shown that the length of the document to which revisions are to be made does not have to be very long to justify the editing capabilities. As the length of the document increases, the benefits of the editing capabilities increase in proportion. It has also been shown that the average user can become quite productive within the first day, and would be most efficient after about six months. These productivity increases can be sufficiently high to completely justify the text processing system by the amount of labor saved.

CHAPTER 6

CONCLUSIONS

Text processing is the application of computer technology to the production of textual information. In the past, computers have been used primarily for numerical work exclusively, but as costs of computers and electronics come down while labor costs continue to rise, there is more incentive for the use of text processing.

There are many text processing systems available and in use today. They range in capacity from automatic form letter generators to vehicles for new and innovative ideas in information dissemination. At the bottom of the scale are the automatic typewriters which have relatively small memory capacities which in turn limit the size or complexity of documents which can be produced easily. They are also limited in their editing and formatting features which provide for only minor changes. Using their storage media for permanent storage can be relatively expensive when compared to magnetic tape on large computer systems, but the cost differential would probably not justify the acquisition of a computer. With some of the systems, substantial efforts have been made to make these typewriters completely conversant with computers so that, in effect, they can serve as I/O devices for that computer in text processing work.

In the middle range of capabilities are the many computer based systems, including minis, time sharing services, and multipurpose computers. This middle range actually covers quite a wide range in capabilities, but minimally, these systems have, or can have large memories, extensive editing capabilities, and complete formatting capabilities. Magnetic tape provides an inexpensive means for permanent storage, which is also less bulky than hardcopy, thereby saving storage space as well. Most computer systems can use a great variety of inputs and outputs ranging from optical character recognition input to photo composition output adding to their flexibility.

Integrated text processing systems are in the upper range of capabilities with today's technology, and are limited in their current applications. Minimally, these systems consist of (1) a data management system, (2) a context editor, (3) document retrieval, (4) a generalized text description language, and (5) a fully modular approach in implementation. This type of system has only been approximated by a few firms to date, but it potentially is a system which will allow many new and innovative methods of providing information in the future.

Most of the present day uses of text processing systems are merely the automation of the more repetitive typing chores of an office. However, some engineering firms use text processing extensively in their documentation for federal regulatory agencies, and production of specifications and contracts. In some cases, the specification can be written by merely supplying some key parameters to a program which logically combines blocks of text for the first draft. While these applications do a great deal to relieve typing loads, one of the prime benefits realized is the time saved by expensive professionals, freeing them for more productive work than cutting and pasting. Manuals, especially program manuals, are also produced by programmers on line. This results in more extensive and precise documentation of programming efforts than existed with conventional means. Where text processing terminals are plentiful, some companies have used the facilities to transmit documents to other divisions or companies for faster and more reliable service than the mail provides. In other situations, they have been used to provide sensitive, but useful information to employees by their ability to access a file with the terminal. Text processing systems are also used for information distribution on the ARPA computer network. These are just a few of the potential uses of text processing systems.

The justification that the different companies give for using text processing varies as much as the uses. Some merely justify the full cost of the system by the stenographic labor actually saved, and others justify the incremental costs of the system by the labor savings. Others view the system as providing benefits that cannot otherwise be had. It can also be justified simply as a morale booster, since it eliminates much of the dull, repetitive jobs faced by both professionals and clericals. The method of justification usually has something to do with the organization of the system. If it is merely a dollar saving which is being sought, then there is a motivation to centralize the facilities and tightly control the efficiency of its operation. If the system is used as a morale booster, then the primary goal would be to provide the system's services to anyone who wished to use them. There have been examples of these and combinations of these justifications in many of the examples investigated.

Whatever the justification, one fact that seems clear is that text processing will increase the productivity of anyone connected with the production of a document. This increase in authors' productivity will result from less time spent proofreading, less time spent cutting and pasting, and with advanced systems, less time spent researching. It is hard to put a performance increase figure on this element, since it is very situation dependent. However, measuring typists' increase in productivity is a little more straightforward since most typing is more or less the same. Several sources from the literature quote increases in productivity ranging from 50 per cent to 1000 per cent. However, these figures include the effects of the whole system, including the benefits derived from the ability to copy a large amount of one document from a previous edition etc. This phenomenon is also situation dependent, so a test was devised to determine the increases due solely to increases in typing efficiency. With ten subjects of varying abilities, the average increase in net words per minute was 58 per cent. Most of the increase was a result of the ability to easily correct mistakes, and type at "draft speed" since there was no fear of making errors.

Some attempt was also made to determine how effective a person can be in exploiting the main benefits of text processing systems, the editing capability. This test showed that after ten minutes of instruction, on the average, it would be faster to make three typical corrections to documents of 117 words or greater than to retype it. Extrapolating to a proficient terminal operator, this cutoff length is about 65 words. Since most documents are considerably longer than this, it would follow that significant increases in productivity can be realized, even initially, from the editing features of text processing systems.

BIBLIOGRAPHY

1. Atomic Energy Commission, "A Study of 4 Methods of Entering Text into Computers." Oak Ridge Nuclear Laboratory report, ORNL TM 3600, Nov. 1, 1971.

2. Auerbach Time Sharing Reports. 40.180, Aug 1969.

3. "An Introduction to Documate." Index Systems Inc., 1973.

4. "Documate--An Abridged Functional Description." Index Systems Inc., 1973.

5. Gildenburg, Robert F. "Word Processing." Modern Data, Jan. 1973.

6. Gilford, Michael. "Second Generation Computer Specifications." Progressive Architecture, May 1972.

7. Goldfarb, Charles J.; Mosher, Edward J.; and Peterson, Theodore I. "Integrated Text Processing for Publishing and Information Retrieval." IBM Cambridge Scientific Center Report, form number G320 2065, April 1971.

8. Kleinschood, Walter A. "The 'Gal Friday' is a Typing Specialist Now." Administrative Management, 1971.

9. "Magnetic tape Selectric Typewriters." Form numbers 543-0506-1, G543-0507-3, IBM Corp., New York.

10. Mauro, Anthony. "Get Your Secretary Away from the Typewriter." Office Equipment and Methods, Aug. 1971.

11. "NSCRIPT User's Guide." AP-55 rev. 1; MIT Information Processing Center, Dec. 14, 1972.

12. "Redactor Editing Typewriters." Forms RMS-7203-10RI, RMS-7108-9, and RMS7112-1RI, Redactron Corporation.

13. Rosen, Harold J. "Computer Specification Programs." Progressive Architecture, Sept. 1971.

14. "SCRIPT User's Manual." IBM Cambridge Scientific Center Report, Form No.320 2053, Cambridge, Mass., 1969.

15. "SCRIPT/370 Program Description/Operations Manual." IBM Form No. sh20-1114-0, IBM Corp., New York., 1973.

16. "Stone & Webster Administrative Terminal System User's Manual." Stone & Webster Inc., Boston, Mass., 1971.

17. Sussman, Joseph M.; Lodcher, Robert D.; and Howard C. Stotland. "SPECS--A Specification Production Computer System." Journal of the Structural Division Proceedings of the American Society of Civil Engineers, Jan. 1971.

18. "System 360 Administrative Terminal System." Applications Description Manual GH20-0297-3; Terminals Operations Manual GH20-0589-2; Terminal Operator's Quick Reference Guide G320-0956-2., IBM Corp., New York.

19. van Dam, Andries; and Rice, David E. "On line Text Editing: a Survey." Computing Surveys Vol. 3 No. 3, Sept. 1971.

20. van Dam, Andries. "Fress--Concepts and Facilities for the Layman." Text Systems Inc., Barrington, R.I., 1972.

21. van Dam, Andries. "Fress User's Guide--Part A & B." Text Systems Inc., Barrington, R.I., 1972.

22. van Dam, Andries, Elliot, W.D., Potas, W.A. "Computer Assisted Tracing of Text Evolution." Fall Joint Computer Conference, 1971.

23. "Wang System 1200 Cassette Typewriter." Forms 700 0493A 11-72-50M; 700 0835-11-72-20M; 700 0493-6-72-50M. Wang Laboratories Inc.

24. White, Donald. "Computer, Take a Letter." Boston Evening Globe, Feb. 14, 1973.

APPENDIX 1
AUTOMATIC WORD PROCESSING TYPEWRITERS

COMPANY/ MODEL	STORAGE MEDIA	EDITING & OTHER FEATURES
American Automatic Typewriter Auto-typist	player-type paper tape roll	manual typing of inside address & salutation; right margin control as option; typewriter input/ output
CPT Corporation CPT 4200	dual cassette with 200-character line buffer	char & line erase; auto right margin adjust; tape-to tape dup; forward & reverse search; word, line, paragraph & page control; Selectric I/O
IBM Office Products Division MC/ST	magnetic card	char & line erase; card track select; CPU/communications interfaces; Selectric input/ output
MT/ST	single or dual cartridge	char & line erase; auto right margin adjust; tape-to-tape dup; forward & reverse search; auto line expand; Selectric & composer I/O
Lexitron Corporation Videotype-1	single cassette with 7000-char CRT & line printer buffers	intelligent CRT display terminal for input & text edit; char & line erase; auto right margin adjust; word, line, paragraph & page control typewriter or printer output
MetroTel Corporation	player&type paper tape roll	roll-controlled auto type unit with converter interface for user supplied typewriter
Mate Series Model 801	single cassette	manual type of corrections & revisions after initial typing; Royal 970 typewriter input/output

Novar/GTE	dual tape loop cartridge with line buffer	char & line erase; tape-to-tape dup; CPU/communications interfaces; Selectric typewriter input/output
QuinData QuinType Model 70	dual paper tape	char & line erase; auto margin adjust; tape-to-tape dup; CPU/ communications interfaces; Selectric I/O
QuinType Model 80	single or dual cassette	char & line erase; auto margin adjust; tape-to tape dup; forward/reverse search; word, line, paragraph, & page control; CPU/ communications interfaces; Selectric input/output
Remington Rand/Office Machines & Redactron Corporation MC100/200	single or dual magnetic card	char & line erase; card track select; card-to-card dup; word, line, paragraph, & page control; Selectric I/O
MT100/200	single or dual cassette	char & line erase; tape-to-tape dup; word, line, paragraph, page control; Selectric typewriter input/output
Ricoh of America Model 6000	dual paper tape	char, word & line stop/ skip; right margin control; tab memory; Selectric typewriter input/output
Savin Business Machines Word Master	single cassette	cassette-transport & controller with interface for user-supplied typewriters
Singer Business Machines Flexowriter	dual paper tape with tab card reader as option	plug-board programmable control; auto address, word, line & paragraph insert; typewriter I/O

SYS Computer Editerm	single or dual cassette with internal memory	intelligent CRT display terminal for input & text edit; char & line erase; margin adjust; word, line,paragraph & page control; Selectric output
Terminal Equipment Edityper	paper tape	auto margin control, line expand, edit, read, skip & tab controls; selectric typewriter input/output
Ty-data Inc. Series 3600	single or dual cassette	auto locate, correct, delete & insert of char, words, lines or paragraphs; auto merge; auto margin adjust; auto message search; Selectric input/output
Wang Labora-tories System 1200	single or dual cassette	auto word-in-line, line, paragraph & page search; auto head centering; auto right margin justify; auto decimal point align; Selectric I/O; CPU & communications interfacing
Word Processing Inc. Scribona 100	5000 charac-ter internal memory cas-sette	auto search; auto char, word,line or paragraph select, skip or de lete; right margin adjust; Selectric I/O
Scribona 500	cartridge	auto search on code to 500 ad-dress locations for insert/delete operations; char correct/erase; word, line or paragraph skip; Selectric I/O

APPENDIX 2

MINICOMPUTER WORD PROCESSING AND FORMATTING SYSTEMS(5)

COMPANY/ MODEL	STORAGE MEDIA	APPLICATIONS & FEATURES
Datatype Corporation Setype	Inputs: document-to-OCR terminal Outputs: paper tape	document-to-6-level paper tape editing & converter system; uses standard Selectric with OCR font to type input document
ECRM Inc. Autoreader	Inputs: document-to-OCR terminal Outputs: paper tape	document-to-6-level paper tape editing & converter system; uses Courier-12 font typewriter to prepare input document
Edit Systems TextEd	Inputs: multiple type-writer terminal Outputs: hardcopy, cassette, mag & paper tape	time-shared text editing, typing & formatting system; typewritten & line printer produced hardcopy; 9-track mag & 6/8-level paper tape out put; cassette document storage
Graphic Systems System 1	Inputs: auto typewriter produced cassettes Outputs: photocomp	text-editing photocomposer system using the Redactron word processing typewriter
Hendrix Electronics Text Publishing System	Inputs: multiple CRT display terminals Outputs: hard-copy, mag & paper tape	stand-alone & time-shared text editing, printing & formatting systems; line printer hardcopy; OCR input option; auto hyphenation, classified ads & business packages

Imlac Corporation Composer 15	Inputs: CRT terminal Outputs: paper & magnetic tape	stand-alone text edit & formatting system; 6-level paper & 7/9-track magnetic tape output
Index Systems Documate	Inputs: multiple typwriter terminal Outputs: hardcopy, mag & paper tape	time-shared text editing, typing & formatting system; typwritten & line printer produced hardcopy; extensive file management capabilities; removable disk storage
Information Control Systems Astrocomp	Inputs: multiple CRT & Selectric terminals Outputs: hardcopy MTST, mag & paper tapes	time-shared text editing, typing & formatting system; typewritten & line printer produced hardcopy; 6/7/8-level paper, 9-track mag and MTST cartridge tape output; auto hyphenation/justification package
LCS Corporation CompuText	Inputs: multiple CRT & Selectric terminals Outputs: hard copy	time-shared text editing & typing system; Selectric typed & line printer produced hardcopy; multi-task, disk file system
MGA Computer 952	Inputs: CRT terminal Outputs: hardcopy	text editing, typing & business processor terminal; serial "daisy" element printer produced hardcopy
Omnitext Inc. Omnitext	Inputs: CRT terminal Outputs: paper tape	text editing & formatting system; 6/8-level paper tape output

SYS Computer DataVerter	Inputs: CRT terminal Outputs: hard-copy & magnetic tape	text editing, typing & formatting system; line printer hard copy; mag tape output
Tal-Star Computer Systems T1000	Inputs: CRT & document-to-OCR terminal Outputs: paper tape	text editing & formatting system; justification & classfied ads packages; paper tape output
Varian Data Machines Varitext	Inputs: multiple Selectric typewriters Outputs: hard-copy,cassette & mag tape	time-shared text editing & typing system; Selectric typed & line printer produced hardcopy; cassette document storage; mag tape output